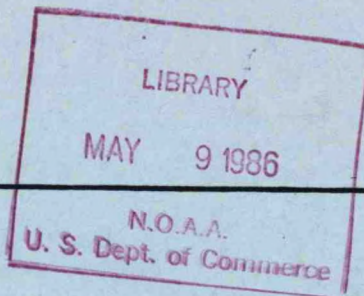


H
QC
874.3
U63
no.37

OAA Eastern Region Computer Programs
and Problems NWS ERCP - No. 37



Hydrograph
Convective Parameters & ~~Hydrograph~~ Program - CONVECT

Hugh M. Stone
National Weather Service Eastern Region
Garden City, New York

Scientific Services Division
Eastern Region Headquarters
April 1986

U.S. DEPARTMENT OF
COMMERCE

National Oceanic and
Atmospheric Administration

National Weather
Service

NOAA TECHNICAL MEMORANDUM

National Weather Service, Eastern Region Computer Programs and Problems

The Eastern Region Computer Programs and Problems (ERCP) series is a subset of the Eastern Region Technical Memorandum series. It will serve as the vehicle for the transfer of information about fully documented AFOS application programs. The format ERCP - No. 1 will serve as the model for future issuances in this series.

- 1 An AFOS version of the Flash Flood Checklist. Cynthia M. Scott, March 1981. (PB81 211252).
- 2 An AFOS Applications Program to Compute Three-Hourly Stream Stages. Alan P. Blackburn, September 1981. (PB82 156886).
- 3 PUPPY (AFOS Hydrologic Data Reporting Program). Daniel P. Provost, December 1981. (PB82 199720).
- 4 Special Search Computer Program. Alan P. Blackburn, April 1982. (PB83 175455).
- 5 Conversion of ALEMBICS Workbins. Alan P. Blackburn, October 1982. (PB83 138313).
- 6 Real-Time Quality Control of SAOs. John A. Billet, January 1983. (PB83 166082).
- 7 Automated Hourly Weather Collective from HRR Data Input. Lawrence Cedrone, January 1983 (PB83 167122).
- 8 Decoders for FRH, FTJ and FD Products. Cynthia M. Scott, February 1983. (PB83 176057).
- 9 Stability Analysis Program. Hugh M. Stone, March 1983. (PB83 197947).
- 10 Help for AFOS Message Comp. Alan P. Blackburn, May 1983. (PB83 213561).
- 11 Stability and Other Parameters from the First Transmission RAOB Data. Charles D. Little, May 1983. (PB83 220475).
- 12 TERR, PERR, and BIGC: Three Programs to Compute Verification Statistics. Matthew R. Peroutka, August 1983. (PB84 127521).
- 13 Decoder for Manually Digitized Radar Observations. Matthew R. Peroutka, June 1983. (PB84 127539).
- 14 Slick and Quick Data Entry for AFOS Era Verification (AEV) Program. Alan P. Blackburn, December 1983. (PB84 138726).
- 15 MDR--Processing Manually Digitized Radar Observations. Matthew R. Peroutka, November 1983. (PB84 161462) (Revised June 1985, PB85-220580/AS)
- 16 RAMP: Stability Analysis Program. Hugh M. Stone, February 1984.(PB84 161447)
- 17 ZONES. Gerald G. Rigdon, March 1984. (PB84 174325)
- 18 Automated Analysis of Upper Air Soundings to Specify Precipitation Type. Joseph R. Bocchieri and Gerald G. Rigdon, March 1984. (PB84 174333)

(Continued on Inside Rear Cover)

EASTERN REGION COMPUTER PROGRAMS AND PROBLEMS - No. 37

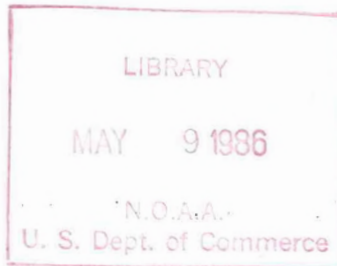
H
QC
874:3
U63
no. 37

Convection Parameters and Hodograph Program - CONVECT

"

Hugh M. Stone
National Weather Service Eastern Region
Garden City, New York

Scientific Services Division
Eastern Region Headquarters
April 1986



ERCPC No. 37
April 1986

Convection Parameters and Hodograph Program - CONVECT

Part A: Program Information and Installation Procedure

Program Name: CONVECT.SV

AAL ID:

Revision No.: 01.00

Function: Computes energy index and other parameters useful in forecasting convection and produces wind hodographs for selected stations.

Program Information:

Development Programmer:

Hugh Stone

Location: ERH, Garden City, NY

Phone: FTS 649-5443

Language: DG FORTRAN IV/5.57

Save File Creation Date: 4/14/86

Running Time: Varies according to number of stations decoded by TTBBDD. Each raob takes approximately 12 seconds to compute convection parameters and an additional 12 seconds, if a hodograph is produced.

Maintenance Programmer:

Hugh Stone

Type: Overlay

Disk Space:

CONVECT.SV	119 RDOS blocks
CONVECT.OL	24 RDOS blocks
TTBBSTAS.DB & STN	0.5 RDOS blocks
Significant level data	7.4 RDOS blocks/10 raobs
Mandatory level data	1.5 RDOS blocks/10 raobs

Program Requirements

Program Files:

Name

Comments

CONVECT.SV

CONVECT.OL

Data Files:

Name

DP Location

Read/Write

Comments

FILNAM.TX

DPO

R

Contains name of primary data file

MMDDYYU.HH (variable)	DPO	R	Primary data file from TTBBBD.
MMDDYYS.HH (variable)	DPO	R	Supplementary data file from TTBBBD.
TTBBSTAS.DB	DPO	R	List of decoded raobs
STN	DPO	R	List of raobs for hodograph production.
INDEXX	DPO	R/W	Temporary
HMSGPH.01	DPO	R/W	Temporary

AFOS Products:

<u>ID</u>	<u>Action</u>	<u>Comments</u>
EISTAB		Table of convection parameters
NMCGPHEIS		Plot of EL/EI
NMCGPHEIT		Plot of 1 or 2 columns of data from EISTAB
NNNHO		Hodographs, NNN is identifier, (10 hodographs possible, with 10 different NNN's)

Load Line:

```

RLDR CONVECT RANN2 CCL1 INDX1 BNDX WOBF SATLFT TCONOF
WMROF DPTOF VAPFW HEIGHT JREAL ISCR5 GPT2 TXX
RSTND BNSCH MTITL GPT1 DIAM SDUV
[PR LIM, SHEAR RHOC PRES CNLMT, FINH GPT]
OUT TOP.LB AG.LB UTIL.LB FORT.LB

```

Program Installation:

1. Add EISTAB to the data base.
2. Add NMCGPHEIS and NMCGPHEIT to the data base and assign map background B02.
3. Add NNNHO to the data base, where NNN is 3 letter station identifier. Up to 10 NNNHO's may be added.
4. Compose the STN file at an ADM console. This is a list of 3 letter station identifiers for which hodographs are wanted. Each identifier must terminate with a carriage return and must be in the same relative order that it appears in the TTBBSTAS.DB file. If less than 10 hodographs are wanted, the

remainder of the list must be filled with zeros "000". If no hodographs are wanted, the entire list must filled with zeros, with carriage return after each third zero. This file should be on DPO or linked to DPO.

5. CONVECT.OL and CONVECT.SV should be on APPL1 of DZ0 and linked to DPO.
6. Make sure that TTBBB.SV (Version 05.00) is installed. Prior versions of this program cannot be used.

Convection Parameters and Hodograph Program - CONVECT

Part B: Program Execution and Error Conditions

Program Name: CONVECT.SV

AAL:ID

Revision No.: 01.00

Program Execution

1. Prior to running CONVECT, the TTBBB upper air decoder (Version 05.00) must be run without using the optional 'M' or 'S' switches, e.g., at ADM type:

RUN:TTBBB

2. To run CONVECT, at an ADM type:

- (a) RUN:CONVECT ,or
- (b) RUN:CONVECT II ,or
- (c) RUN:CONVECT IIJJ

The II and JJ in (b) and (c) refer to column numbers of the EISTAB product, which specifies the parameter(s) to be plotted on the EIT graphic. For example, the command,

RUN:CONVECT 0905

produces an EIT graphic with mixing ratio WAVG (column 9) on the left side of the station circle and maximum parcel level MPL (column 5) on the right. Only 2 or 4 numerals may appear after CONVECT on the command line. II and JJ must correspond to column numbers of the EISTAB product, i.e., they must be numerals between 01 and 12. If no numerals are specified on the command line (a), then EIT will be blank.

3. When CONVECT begins, it types on the Dasher the name of the decoded upper air data file, which contains the month, date, year, and hour of the data, and the number of hodographs to be produced. As the program runs, numerals appear on the Dasher, which tell how far the computation has progressed down the list in the TTBBSTAS.DB file. For example, if the eighth raob is being processed, '8' appears on the Dasher, but, if this raob will have a hodograph '-8' appears on the Dasher.

Error Conditions

Error messages to the ADM are of the form:

JOB CONVECT ABORTED - ERROR CONDITION : MESSAGE

where MESSAGE is given in the following list. Execution is terminated whenever these occur:

ADM Messages

Meaning

- | | |
|------------------|--|
| 1. TTBBSTAS.DB | Cannot read TTBBSTAS.DB file.
Probable system problem. |
| 2. NOT IN STDIR | Raob station being processed is
not in STDIR.MS file. Station
ID is written to the Dasher. |
| 3. BAD HODO LIST | List of hodograph stations in
file STN could not be read.
Probable system problem. |

Dasher messages

- | | |
|--------------------------|---|
| 1. CMD LINE ERR, L = ... | Numerical entry on command line
is incorrect. Only 2 or 4
numerals may be entered, and
column no. specified must not
exceed 12. L = No. of numerals
entered + 1. |
| 2. NNN NOT IN STDIR | Station NNN not found in STDIR
file. (Accompanies ADM message
2 above.) |
| 3. ERROR IN CONVECT | Indicates logical error in
CONVECT. Should never occur. |
| 4. ERROR IN CCL1 | Logical error in CCL1.
Should never occur. |
| 5. ERROR IN CNLMT | Logical error in CNLMT.
Should never occur. |

All of the above error messages are followed by a program termination.

Convection Parameters and Hodograph Program - CONVECT

Hugh M. Stone
Scientific Services Division
National Weather Service Eastern Region
Garden City, New York

I. Introduction

This program computes many parameters which are useful in forecasting convection, such as several stability indices, including the energy index (EI), height of the equilibrium level (EL), maximum parcel level (MPL), etc. The program is similar to the RANP program (Stone, 1984) and many parameters computed by RANP are also output from CONVECT. Some items from RANP, which have demonstrated little or no usefulness, i.e., ETCCL, EI2, and parameters related to potential instability (Stone, 1985), are not computed by CONVECT, and have been replaced by parameters believed to be more promising.

The EI1 index of the RANP program is now simply called EI in the CONVECT program, since EI2 is no longer computed. The EI index is defined as the change of kinetic energy of an entraining parcel as it moves upward from a low level of the atmosphere to the 400 millibar level; the entrainment rate used increases the mass of the parcel by 60 percent over a 500 millibar ascent. Entrainment is also used in the computation of the lifting condensation level LCL and the level of free convection LFC. All other parameters are computed without entrainment.

CONVECT also provides wind hodographs to 6 kilometers above ground level for selected upper air stations. Recent research indicates that the hodograph is useful in determining the character of convection and its direction of motion after it develops (Weisman and Klemp, 1984). This is relevant not only to severe weather forecasting but also to the heavy rainfall problem. Numerical studies have shown that, given the same amount of instability, the type of convection that develops strongly depends on the wind shear structure of the lowest 6 kilometers of the atmosphere. For example, short-lived multicell convection develops with weak wind shear, while stronger convection requires more wind shear. Wind shear vectors veering with height favor the development of right-moving, cyclonically rotating storms. Wind shear structure is easily determined by use of the hodograph.

The input to the CONVECT program comes from the decoded upper air data files provided by the TTBBBD program (Stone, 1985). Mandatory and significant levels are combined by the TTBBBD program, so a better assessment of stability is possible, rather than using significant levels only, such as the RANP program does. TTBBBD can also decode the Canadian raobs which may be important for forecast areas near the Canadian border.

II. Methodology and Software Structure

A. Methodology

Convection parameters are computed for every upper air station listed in file TTBBSTAS.DB, which is used by the TTBBBD program for creating the decoded data files. An example of this file is shown in Fig. 1. TTBBBD produces two data files, one includes significant levels with mandatory levels inserted where required, and the other includes heights of mandatory pressure surfaces and other miscellaneous information. Both data files must be present before running the CONVECT program.

Convective parameters are output to file INDEXX, which is later stored as AFOS product EISTAB (Fig. 3.). This product is similar to the WRKTPC product generated by the RANP program, except that half of the parameters are different. Parameters listed in the twelve columns of EISTAB are as follows:

1. P0MX (mb)

Height above surface of parcel used for EI and EL computations, i.e., $P(0)$ -P_{MAX}, where P_{MAX} denotes pressure of parcel in lowest 150 millibars of atmosphere having highest wet bulb potential temperature.

2. WMAX (m/sec)

Maximum vertical velocity of a non-entraining parcel at the EL. This is computed only if there is a net positive energy area, i.e., sum of negative and positive areas greater than zero, between P_{MAX} and EL. $WMAX = (2 \times NET \text{ POS. AREA})^{1/2}$.

3. EL (mb)

Equilibrium level: base of the uppermost negative energy area encountered by a non-entraining parcel.

4. EL (hundreds of ft)

Equilibrium level converted from pressure to height units by using interpolation between raob mandatory levels, or if missing, U. S. standard atmosphere values are assumed and "E" is appended to indicate an estimated height.

5. MPL (hundreds of ft)

Maximum parcel level: height at which negative energy area above EL balances the net positive area below the EL. It is only computed when sum of positive and negative areas between P_{MAX} and EL is positive. Parcel attains maximum upward velocity at EL and decelerates to a stop at the MPL. The relationship between observed radar tops and MPL may have some significance for severe weather development. A value of -888 denotes that MPL is above 100 mbs and cannot be determined, since we do not have decoded data above 100 mbs. A value of -777 indicates that the raob terminated below 100 mbs and MPL is above highest reported level.

6. TROP (hundreds of ft)
Tropopause height from mandatory level upper air observations converted from pressure to height units.
7. EI+ (j/kg x 10)
Positive part of the energy index, EI.
8. EI- (j/kg x 10)
Negative part of EI. $EI = (EI+) + (EI-)$. There may be some value in considering positive and negative parts separately, since it has been observed that a strong negative energy area in the lower troposphere tends to suppress convection, although EI alone may be positive. Negative energy areas are more commonly observed in the lower troposphere rather than the upper part.
9. WAVG (g/kg)
Average mixing ratio in lowest 100 millibars of the atmosphere. Useful for locating low level moisture maxima, where convection may develop.
10. LI (deg C)
Lifted index.
11. KI (deg C)
K index.
12. SWI (deg C)
Showalter index.

Any parameters in EISTAB that cannot be computed for any reason are indicated by -999. If the raob does not extend to at least 450 millibars, convective parameters are not computed and a message will appear in EISTAB: "SHORT RAOB, P = ...", where P is the lowest pressure reached in the raob. If a raob is completely missing, this is simply indicated by "MISSING".

The EI index, sum of columns (7) and (8), and EL in hundreds of feet, column (4), are plotted on AFOS graphic EIS (Fig. 4.). This is the same as the EIS graphic generated by the RANP program. All graphics are prepared using the AG.LB subroutines (MacDonald, 1981).

An additional AFOS graphic EIT is generated and consists of a plot of any one or two columns of data from the EISTAB product. Column numbers are specified on the command line, when running the program. If none are specified, EIT will be a blank map. An example of the EIT product is shown in Fig. 5., in which mixing ratio WAVG is plotted on the left side of the station circle and maximum parcel level MPL is on the right. If the right hand value is positive, a black station circle is plotted, otherwise that station is indicated by an open circle. This makes it easier to notice positive values on the map. If only one variable is plotted, it will appear on the right side of the station circle. A circle with "M" inside, indicates both variables missing.

Up to 10 wind hodographs may be generated as the convective parameters are being computed. Stations for which hodographs are desired are specified in a list of three character station identifiers stored in RDOS file STN (Fig. 2.). This list of raob stations is a subset of the complete list in file TTBBSTAS.DB. Hodographs extend from the surface to the wind level equal or exceeding six kilometers above ground level, which usually means exceeding 20 thousand feet MSL. An example of a hodograph is shown in Fig. 6.

Convective parameters are listed on the left hand side of the hodograph. Some parameters are listed which are not in the EISTAB product, as follows:

1. B^+ (m/sec)²

The positive energy area that a non-entraining parcel encounters as it is lifted from the level P_{MAX} to EL. Note that the unit (m/sec)² is the same as the unit j/kg, however, EI values have been divided by a factor of 10 to produce a more convenient size number, but units of B⁺ have not been divided.

2. B^- (m/sec)²

The negative energy area that a non-entraining parcel encounters during a lift from P_{MAX} to EL.

3. SHR (m/sec)²

A measure of wind shear over the lowest six kilometers of the atmosphere, defined by the equation:

$$SHR = 1/2 U^2$$

where U is calculated by taking the vector difference between the density weighted mean wind over the lowest 6 kilometers of the atmosphere and the density weighted mean wind from the surface to 500 meters above the ground.

4. BRN

Bulk Richardson Number, defined by the equation:

$$BRN = B^+/SHR.$$

Recent research (Weisman and Klemp, 1984) indicates that the BRN is useful in determining the the type of convection that may develop: single cell, multi-cell, or super cell.

Parameters enclosed in the boxed area are all computed with entrainment that increases the mass of the parcel by 60 percent over a 500 millibar ascent. Parameters outside the boxed area do not involve entrainment.

5. LCL (mb)

Lifting condensation level, where parcel becomes saturated.

6. LFC (mb)

Level of free convection: base of lowest positive energy area, excluding any small positive areas that may result from a superadiabatic lapse rate near the surface.

7. ENERGY CHANGE IN LAYERS (j/kg x 10)

Energy change between indicated P1 and P2 pressures (mb). The lowest eight energy areas are listed beginning at the parcel starting point PMAX. There are usually less than 8 areas for a typical raob, so the listing is usually complete.

8. CCL (mb)

Convective condensation level: on a thermodynamic diagram, this is intersection of mixing ratio line representative of lowest 100 millibars with the temperature trace. If there is more than one intersection, the highest is used.

9. C TMP (deg C & F)

Convective temperature: temperature at CCL level reduced dry adiabatically to surface level. Supposedly indicates temperature at which convection is likely to be initiated.

Hodograph is plotted using U and V wind components at standard levels and mandatory levels. Units are knots and data points are labeled in thousands of feet MSL. Mandatory level winds are decoded in hundreds of feet, then this value is rounded to the nearest thousand for labelling the data point. This sometimes results in two points with the same label on the hodograph. The surface wind is always labelled "0" zero. Large numbers at the ends of the U and V coordinate axes are wind directions, e.g., a positive U component with V = 0, corresponds to a 270 degree wind direction, and a negative U to a 90 degree wind direction. The V axis is similarly labelled. The number of wind levels plotted on the hodograph is indicated by the value "L" in the extreme lower right corner of the graphic.

The hodograph normally extends to the first wind level at or above 6 kilometers above ground level, but data points are labelled in thousands of feet MSL. If the raob does not extend to at least 450 millibars, convective parameters are not computed and the left side of the graphic will be blank but the hodograph will contain whatever wind levels are available. If winds are also missing in this situation, the entire graphic will be blank.

If the radiosonde goes high enough to compute convective parameters, but winds are missing, then these parameters will be listed on the left side of the graphic and the hodograph will be blank. Whatever information is available will be on the graphic. If certain parameters cannot be computed, which frequently happens under very stable atmospheric conditions, then "XXX" is printed for those parameters.

A skew-T plot of the UMN raob is shown in Fig.7; this is the same raob whose hodograph and convective parameters are shown in Fig. 6. Energy areas are shown for a non-entraining parcel rising from the PMAX level. The positive area B+ is used for the Bulk Richardson Number BRN computation, and the net positive energy

below the EL, (B+) + (B-), is the energy that must be balanced above the EL to determine the maximum parcel level MPL.

B. Software Structure

The main program is called CONVECT. Input data are read from the decoded upper air data provided by the TTBBBD program. The computation is done for all upper air data present in the decoded data files. Output is stored in the AFOS Products EISTAB, EIS, and EIT. Up to 10 optional hodographs are stored in the products NNNHO, where NNN is the raob identifier. All graphics are created using subroutines of the AG.LB (MacDonald, 1981).

Three overlays were needed to make the program fit the available memory. All overlays are loaded outside the main loop of the program, so multiple use of the same overlay is avoided. The software structure and load line are shown in Fig. 8. Twenty eight subroutines are used which are briefly described below in approximately the same order that they are used in the program:

PRLIM

Reads numeric code from command line, which specifies variable for plotting on EIT graphic. Opens TTBBBD decoded data files and reads hodograph station file STN.

MTITL

Makes date/time title for graphics: EIS, EIT, and NNNHO.

RSTND

Reads data from station directory file STDIR.MS using BNSCH subroutine. Items used are station elevation and map coordinates for plotting.

BNSCH

Does binary search of data in station directory file. (This subroutine by Rich Thomas, AOD).

CCL1

Computes mean mixing ratio in lowest 100mbs, convective condensation level CCL, and convective temperature C TMP.

INDX1

Computes lifted index, K index, and Showalter index.

BNDX

Determines pressure level PMAX that has highest wet bulb potential temperature in lowest 150 millibars of the atmosphere. If an

identical value is found at two levels, the lowest level (highest pressure) is selected for PMAX. This is the starting level for all parcel calculations and is used because parcel at this level is the most unstable.

RANN2

This subroutine does all energy area computations, which are used to determine the energy index, equilibrium level, lifting condensation level, level of free convection, etc. CONVECT calls this subroutine twice; the first call computes equilibrium level without entrainment and the second call computes the other variables with entrainment.

DPTOF SATLET TCONOF VAPFW WMROF WOFB

All thermodynamic computations are done using these six subroutines from the Nation Severe Storms Forecast Center, Kansas City, MO. (Doswell, et al., 1982).

CNLMT

Finds maximum parcel level MPL, where negative energy area above the EL balances the net positive energy below the EL. Theoretically, this is the level at which an ascending parcel decelerates to a stop.

HEIGHT

Computes height of an arbitrary pressure by interpolating or extrapolating from the pressure/height relationship of standard mandatory levels including station elevation and surface pressure.

JREAL

Rounds a floating point number to an integer.

SHEAR

Computes density weighted mean wind from surface to 6 kilometers above ground level and density weighted wind from surface to 500 meters AGL. From these two items shear factor SHR is computed for use in Bulk Richardson Number BRN.

PRES

Inverse of the HEIGHT subroutine. Computes pressure for a specified height in meters using the same pressure/height relationship.

RHOC

Computes densities for an array of pressures, given the levels of a raob.

SDUY

Converts wind speed and direction to U and V components.

GPT2

Writes convective parameters on left side of the hodograph, then calls GPT1 to draw the hodograph, and finally, stores the hodograph into AFOS product NNNHO.

TXX

Converts a real number to integer and writes it to a specified location on a graphic. Used in GPT2 subroutine.

ISCR5

Converts a 1 to 5 digit integer to ASCII characters in an array. For use with the TEXT subroutine.

GPT1

Plots hodograph to first reported wind level at or above 6 kilometers above ground level.

DIAM

Draws diamond shaped box around a data point on a graphic.

FINH

Inserts heading and ending on file INDEXX and stores it into AFOS graphic EISTAB. Calls GPT subroutine to create EIS and EIT graphics.

GPT

Plots specified variables onto map background B02. Used for EIS and EIT graphics.

III. Cautions and Restrictions

The program cannot process more than 50 raob stations at a time, so the TTBBSTAS.DB file must be restricted to this number. The file STN that contains the list of stations for hodograph production is limited to 10 stations and they must appear in the same relative order that they appear in TTBBSTAS.DB. For example, if hodographs are wanted for the 3rd, 7th, and 19th stations of the TTBBSTAS.DB list, then they must be listed in the order 3, 7, 19 in file STN. The file STN must be edited at an ADM with each 3 letter identifier terminating with a carriage return. Exactly ten entries must appear in STN, so if less than 10 hodographs are wanted, the remaining spaces must be filled with zeros, Fig. 2. If no hodographs are wanted STN must contain all zeros.

It is important that correct station elevation be read from the station directory file STDIR.MS. This should be the station elevation of the WSMO and not the WSFO or WSO. If the wrong elevation is used, the program will still run, but the wind shear factor SHR used to compute Bulk Richardson Number BRN will be incorrect, and the low level pressure/height relationship will be slightly erroneous.

It is unlikely that the CONVECT program will be used along with RANP, since there is much redundancy in the output; however, both programs produce the EIS graphic and they look very similar. The EIS graphic produced by RANP has the entrainment rate "EFF" contained in the title, whereas, that produced by CONVECT does not have it.

IV. References

Doswell, C.A., J. T. Schaefer, D. W. McCann, T. W. Schlatter, and H. B. Wobus, 1982: Thermodynamic Analysis Procedures at the National Severe Storms Forecast Center, Preprints, 9th Conf. Weather Forecasting and Analysis, Amer. Meteor. Soc., Seattle, 304-309.

MacDonald, A.E., 1981: AFOS Graphics Creation from FORTRAN, NOAA Western Region Computer Programs and Problems NWS WRCP-No. 18, National Weather Service, Salt Lake City, UT.

Stone, H.M., 1984: RANP: Stability Analysis Plot Program, NOAA Eastern Region Computer Programs and Problems NWS ERCP-No. 16, National Weather Service, Garden City, NY.

Stone, H.M., 1985: A Comparison Among Various Thermodynamic Parameters for the Prediction of Convective Activity - Part II, NOAA Technical Memorandum NWS ER-69, National Weather Service, Garden City, NY.

Stone, H.M., 1985: TTBBB, Significant and Mandatory Level Decoder, Eastern Region CP Brief No. 12, National Weather Service, Garden City, NY.

Weisman, M.L. and J. B. Klemp, 1984: The Structure and Classification of Numerically Simulated Convective Storms in Directionally Varying Wind Shears, Mon. Wea. Rev., 112, 2479-2498.

V. Program Information and Procedures for Installation and Execution

Convection Parameters and Hodograph Program - CONVECT

Part A: Program Information and Installation Procedure

Program Name: CONVECT.SV

AAI ID:

Revision No.: 01.00

Function: Computes energy index and other parameters useful in forecasting convection and produces wind hodographs for selected stations.

Program Information:

Development Programmer:
Hugh Stone

Maintenance Programmer:
Hugh Stone

Location: ERH, Garden City, NY
Phone: FTS 649-5443

Language: DG FORTRAN IV/5.57

Type: Overlay

Save File Creation Date: 4/14/86

Running Time: Varies according to number of stations decoded by TTBBDD. Each raob takes approximately 12 seconds to compute convection parameters and an additional 12 seconds, if a hodograph is produced.

Disk Space:

CONVECT.SV	119 RDOS blocks
CONVECT.OL	24 RDOS blocks
TTBBSTAS.DB & STN	0.5 RDOS blocks
Significant level data	7.4 RDOS blocks/10 raobs
Mandatory level data	1.5 RDOS blocks/10 raobs

Program Requirements

Program Files:

Name

Comments

CONVECT.SV
CONVECT.OL

Data Files:

Name

DP Location

Read/Write

Comments

FILNAM.TX

DPO

R

Contains name of primary data file

MMDDYYU.HH (variable)	DPO	R	Primary data file from TTBBBD.
MMDDYYS.HH (variable)	DPO	R	Supplementary data file from TTBBBD.
TTBBSTAS.DB	DPO	R	List of decoded raobs
STN	DPO	R	List of raobs for hodograph production.
INDEXX	DPO	R/W	Temporary
HMSGPH.01	DPO	R/W	Temporary

AFOS Products:

<u>ID</u>	<u>Action</u>	<u>Comments</u>
EISTAB		Table of convection parameters
NMCGPHEIS		Plot of EL/EI
NMCGPHEIT		Plot of 1 or 2 columns of data from EISTAB
NNNHO		Hodographs, NNN is identifier, (10 hodographs possible, with 10 different NNN's)

Load Line:

```

RLDR CONVECT RANN2 CCL1 INDX1 BNDX WOFB SATLFT TCONOF
WMROF DPTOF VAPFW HEIGHT JREAL ISCR5 GPT2 TXX
RSTND BNSCH MTITL GPT1 DIAM SDUV
[PRLIM, SHEAR RHOC PRES CNLMT, FINH GPT]
OUT TOP.LB AG.LB UTIL.LB FORT.LB

```

Program Installation:

1. Add EISTAB to the data base.
2. Add NMCGPHEIS and NMCGPHEIT to the data base and assign map background B02.
3. Add NNNHO to the data base, where NNN is 3 letter station identifier. Up to 10 NNNHO's may be added.
4. Compose the STN file at an ADM console. This is a list of 3 letter station identifiers for which hodographs are wanted. Each identifier must terminate with a carriage return and must be in the same relative order that it appears in the TTBBSTAS.DB file. If less than 10 hodographs are wanted, the

remainder of the list must be filled with zeros "000". If no hodographs are wanted, the entire list must filled with zeros, with carriage return after each third zero. This file should be on DPO or linked to DPO.

5. CONVECT.OL and CONVECT.SV should be on APPL1 of DZ0 and linked to DPO.
6. Make sure that TTBB.D.SV (Version 05.00) is installed. Prior versions of this program cannot be used.

Convection Parameters and Hodograph Program - CONVECT

Part B: Program Execution and Error Conditions

Program Name: CONVECT.SV

AAL:ID

Revision No.: 01.00

Program Execution

1. Prior to running CONVECT, the TTBBDD upper air decoder (Version 05.00) must be run without using the optional 'M' or 'S' switches, e.g., at ADM type:

RUN:TTBBDD

2. To run CONVECT, at an ADM type:

- (a) RUN:CONVECT ,or
- (b) RUN:CONVECT II ,or
- (c) RUN:CONVECT IIJJ

The II and JJ in (b) and (c) refer to column numbers of the EISTAB product, which specifies the parameter(s) to be plotted on the EIT graphic. For example, the command,

RUN:CONVECT 0905

produces an EIT graphic with mixing ratio WAVG (column 9) on the left side of the station circle and maximum parcel level MPL (column 5) on the right. Only 2 or 4 numerals may appear after CONVECT on the command line. II and JJ must correspond to column numbers of the EISTAB product, i.e., they must be numerals between 01 and 12. If no numerals are specified on the command line (a), then EIT will be blank.

3. When CONVECT begins, it types on the Dasher the name of the decoded upper air data file, which contains the month, date, year, and hour of the data, and the number of hodographs to be produced. As the program runs, numerals appear on the Dasher, which tell how far the computation has progressed down the list in the TTBBSTAS.DB file. For example, if the eighth raob is being processed, '8' appears on the Dasher, but, if this raob will have a hodograph '-8' appears on the Dasher.

Error Conditions

Error messages to the ADM are of the form:

JOB CONVECT ABORTED - ERROR CONDITION : MESSAGE

where **MESSAGE** is given in the following list. Execution is terminated whenever these occur:

ADM Messages

Meaning

- | | |
|------------------|--|
| 1. TTBBSTAS.DB | Cannot read TTBBSTAS.DB file.
Probable system problem. |
| 2. NOT IN STDIR | Raob station being processed is
not in STDIR.MS file. Station
ID is written to the Dasher. |
| 3. BAD HODO LIST | List of hodograph stations in
file STN could not be read.
Probable system problem. |

Dasher messages

- | | |
|--------------------------|---|
| 1. CMD LINE ERR, L = ... | Numerical entry on command line
is incorrect. Only 2 or 4
numerals may be entered, and
column no. specified must not
exceed 12. L = No. of numerals
entered + 1. |
| 2. NNN NOT IN STDIR | Station NNN not found in STDIR
file. (Accompanies ADM message
2 above.) |
| 3. ERROR IN CONVECT | Indicates logical error in
CONVECT. Should never occur. |
| 4. ERROR IN CCL1 | Logical error in CCL1.
Should never occur. |
| 5. ERROR IN CNLMT | Logical error in CNLMT.
Should never occur. |

All of the above error messages are followed by a program termination.

INTENTIONALLY
LEFT
BLANK

PHLGLACY
 ATLSGLAHN
 ALBSGLALB
 BHMSGLAQQ
 ATLSGLAYS
 MEMSGLBNA
 BUFSGLBUF
 NEWSGLBVE
 PUMSGLCAR
 BOSSGLCHH
 BHMSGLCKL
 CAESGLCHS
 CLESGLDAY
 DENSGLDEN
 SATSGLDRT
 MIASGLEW
 ARBSGLFNT
 FTWSGLGGG
 MKESGLGRB
 RDUSGLGSO
 RDUSGLHAT
 CRWSGLHTS
 WBCSGLIAD
 JANSGLJAN
 NEWSGLLCH
 LITSGLLIT
 MIASGLPBI
 CHISGLPIA
 PITSGLPIT
 PUMSGLPUM
 SJUSGLSJU
 CHISGLSLO
 ARBSGLSSM
 MSPSGLSTC
 MIASGLTBW
 STLSGLUMN
 SATSGLVCT
 WBCSGLWAL
 WULMANUMJ
 WUXMANWOS
 WUXMANWSA
 WULMANYZV

ACY
 ALB
 BUF
 DEN
 DRT
 UMN
 VCT
 000
 000
 000

Fig.1. Example of TTBBSTAS.DB file,
 list of raobs for which
 convective parameters are to
 be computed. Limit is 50.

Fig.2. Example of STN file,
 list of raobs for
 hodograph production.
 Limit is 10.

RAOB PARAMETERS FOR 4/18/86 12Z DP = 50. EFF = 60. PX = 400.

UNITS: (M) (M/SEC) (M) (...FT X 100...) (J/KG X 10) (G/KG) (...DEG C....)

STN	PRM	UWAX	EL	EL	MPL	TROP	EI+	EI-	WAVG	LI	KI	SWI	STN	
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)			
1	ACY	0.	1811.	1.	-999.	340.	0.	-207.	0.90	20.	-34.	16.	ACY	
2	AHN	150.	840.	52.	-999.	393.	0.	-187.	4.44	16.	2.	11.	AHN	
3	ALB	150.	862.	46.	-999.	359.	0.	-181.	4.33	13.	-12.	11.	ALB	
4	AQQ	56.	961.	14.	E-999.	-999.	0.	-256.	4.59	16.	-12.	13.	AQQ	
5	AYS	110.	903.	33.	-999.	419.	0.	-272.	4.22	19.	-27.	16.	AYS	
6	BNA	115.	884.	39.	-999.	400.	0.	-215.	4.77	17.	-10.	16.	BNA	
7	BUF	146.	850.	50.	-999.	390.	0.	-152.	3.20	14.	-1.	9.	BUF	
8	BVE	18.	1000.	4.	E-999.	-999.	0.	-199.	6.49	9.	6.	10.	BVE	
9	CAR	59.	945.	22.	-999.	329.	0.	-170.	3.07	16.	-7.	14.	CAR	
10	CHH	150.	866.	43.	-999.	349.	0.	-124.	4.76	16.	22.	0.	CHH	
11	CKL	127.	875.	40.	E-999.	-999.	0.	-140.	4.74	17.	3.	10.	CKL	
12	CHS	150.	867.	44.	-999.	371.	0.	-206.	4.81	17.	-2.	12.	CHS	
13	DAY	136.	1.	842.	52.	-999.	306.	0.	-117.	5.57	12.	15.	6.	DAY
14	DEN	126.	0.	507.	141.	-999.	293.	1.	-25.	2.71	3.	-999.	-999.	DEN
15	DRT	106.	52.	179.	410.	E 484.	-999.	24.	-36.	12.59	-4.	9.	2.	DRT
16	EYW	0.	0.	909.	29.	E-999.	-999.	0.	-180.	9.17	0.	-15.	14.	EYW
17	FNT	127.	0.	860.	45.	-999.	367.	0.	-150.	3.60	14.	-9.	10.	FNT
18	GGG	150.	13.	259.	332.	E 377.	-999.	0.	-43.	0.14	4.	7.	-1.	GGG
19	GRB	146.	0.	850.	50.	-999.	390.	0.	-100.	3.31	10.	0.	12.	GRB
20	GSO	150.	0.	836.	52.	E-999.	-999.	0.	-176.	4.30	15.	6.	12.	GSO
21	HAT	0.	17.	762.	75.	136.	313.	12.	-00.	5.32	10.	17.	0.	HAT
22	HTS	64.	0.	927.	26.	-999.	390.	0.	-170.	5.60	10.	-13.	11.	HTS
23	IAD	150.	0.	857.	46.	-999.	361.	0.	-110.	5.00	13.	17.	7.	IAD
24	JAN	110.	0.	889.	36.	E-999.	-999.	0.	-155.	5.74	14.	15.	10.	JAN
25	LCH	14.	3.	943.	20.	E 26.	-999.	0.	-90.	9.45	2.	2.	12.	LCH
26	LIT	97.	0.	899.	32.	E-999.	-999.	0.	-142.	5.69	10.	11.	0.	LIT
27	PBI	37.	0.	979.	9.	E-999.	-999.	0.	-247.	4.90	13.	-23.	16.	PBI
28	PIA	145.	0.	850.	49.	-999.	415.	0.	-101.	5.51	13.	0.	11.	PIA
29	PIT	150.	0.	829.	56.	-999.	376.	0.	-160.	2.61	16.	-14.	11.	PIT
30	PWM	150.	0.	872.	43.	-999.	362.	0.	-202.	3.34	17.	-1.	13.	PWM
31	SJU	MISSING												
32	SLO	133.	0.	867.	44.	-999.	416.	0.	-191.	4.53	17.	-4.	12.	SLO
33	SSM	124.	0.	875.	43.	-999.	390.	0.	-225.	2.51	10.	-16.	15.	SSM
34	STC	145.	0.	833.	53.	E-999.	-999.	0.	-126.	4.75	14.	21.	10.	STC
35	TBW	16.	0.	1000.	4.	E-999.	-999.	0.	-273.	5.39	14.	-22.	15.	TBW
36	UMN	93.	30.	199.	300.	E 452.	-999.	27.	-0.	6.75	10.	37.	-3.	UMN
37	VCT	0.	17.	251.	339.	E 403.	-999.	10.	-63.	12.85	-4.	13.	-3.	VCT
38	WAL	150.	0.	860.	44.	-999.	333.	0.	-01.	5.94	10.	27.	5.	WAL
39	WMJ	150.	0.	855.	49.	-999.	344.	0.	-196.	2.00	10.	-9.	13.	WMJ
40	WOS	131.	0.	800.	37.	-999.	403.	0.	-109.	2.00	22.	-7.	13.	WOS
41	WSA	150.	0.	871.	42.	-999.	365.	0.	-106.	3.19	22.	1.	10.	WSA
42	YZV	150.	0.	876.	42.	-999.	511.	0.	-244.	1.09	21.	-10.	17.	YZV

Fig.3. Example of EISTAB product, table of convective parameters. Missing values indicated by -999, but in addition, for MPL, -888 indicates MPL above 100mbs, and -777 indicates MPL above highest level of raob, which terminates below 100mbs. "E" appended to values in column 4 indicate that U. S. standard atmosphere was assumed in converting pressure to height units and also applies to column 5.

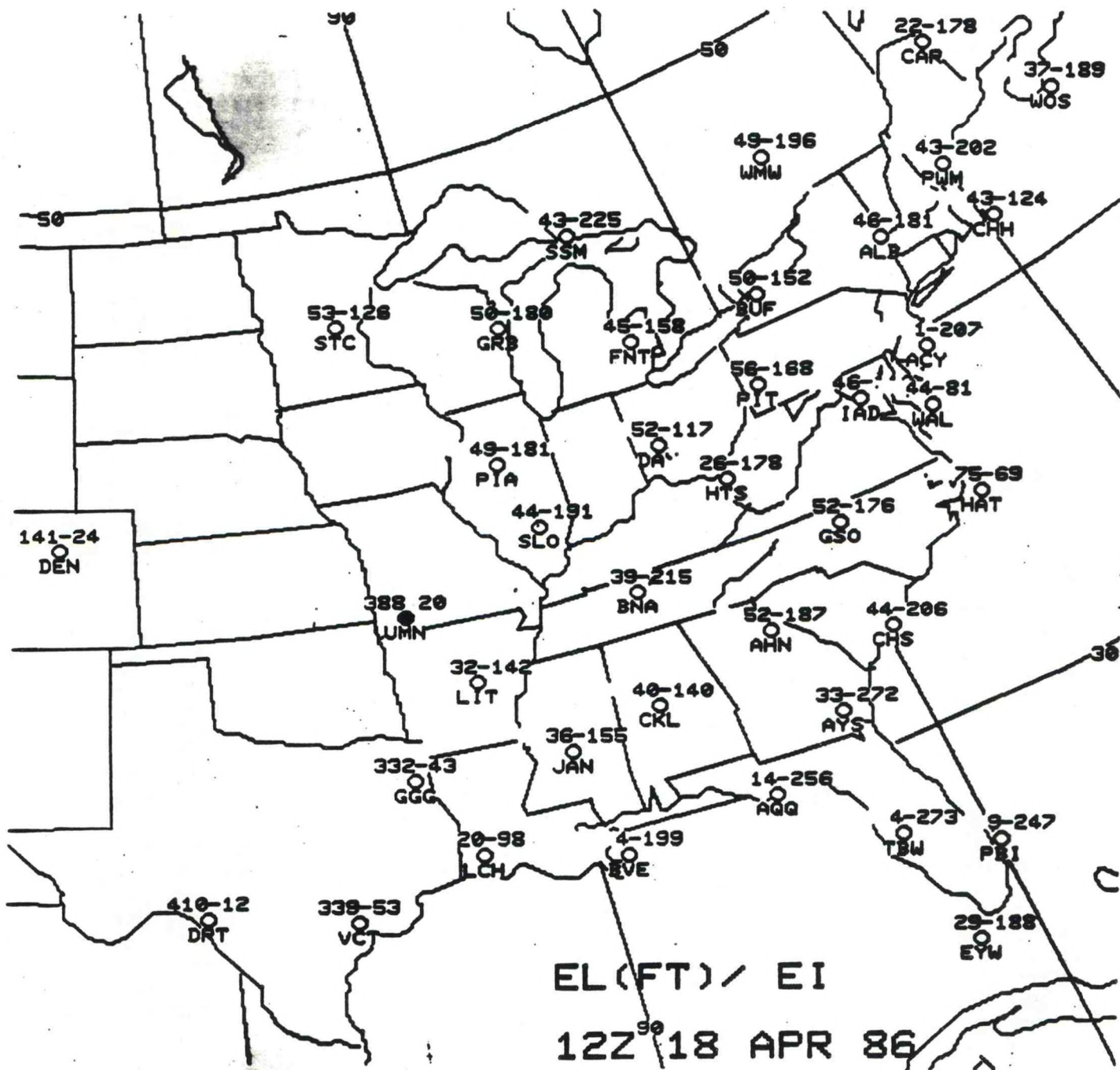


Fig.4. Example of EIS graphic, plot of EL and EI. Positive values of EI are plotted with a solid station circle, for greater visibility. Missing data indicated by a circle with 'M' inside.

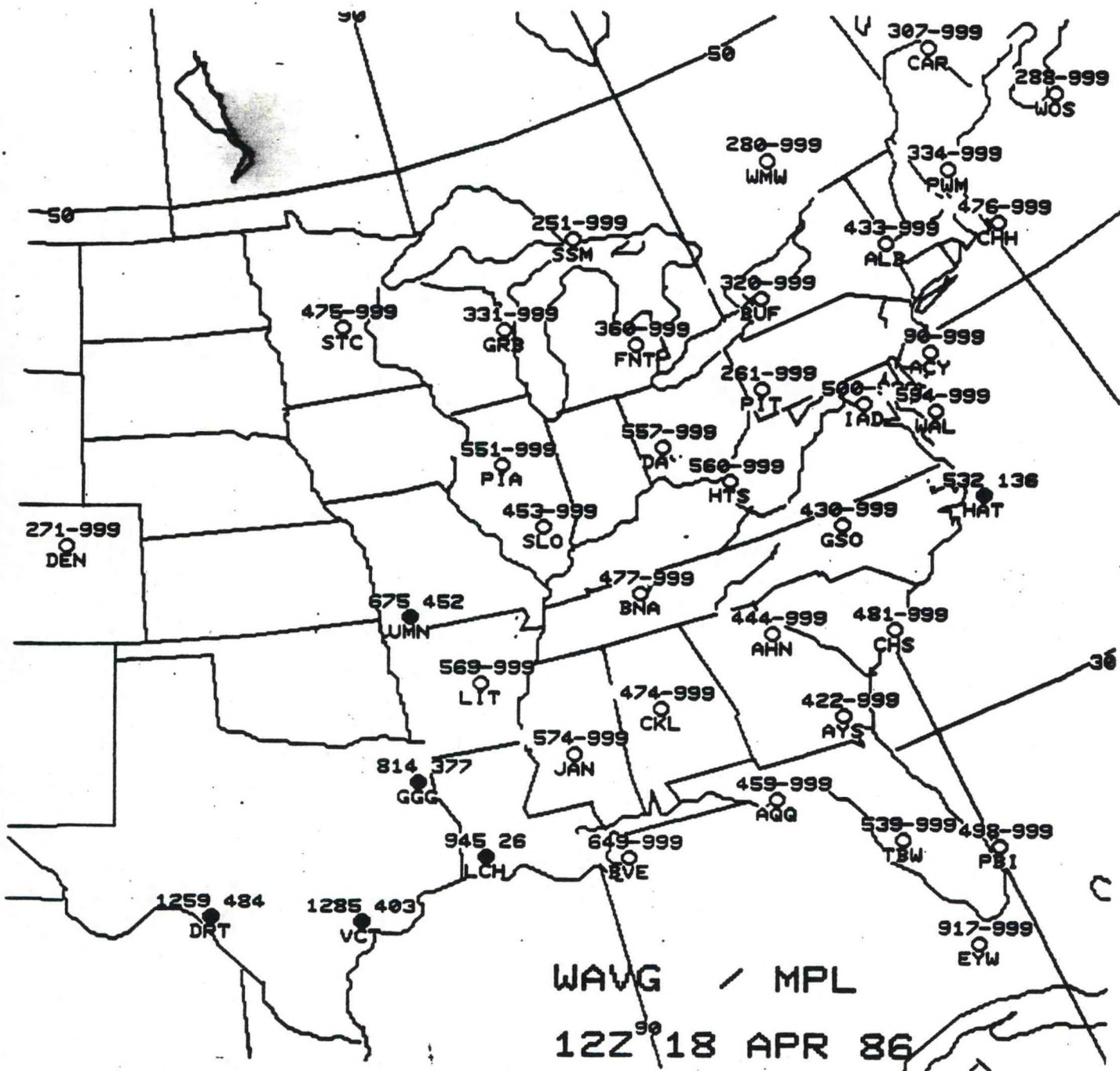


Fig.5. Example of EIT graphic. In this case, column 9 (WAVG) and column 5 (MPL) of the EISTAB product (Fig.3.) are plotted. Values of -999 indicate missing data or could not be computed for some reason. If both values are missing a circle with 'M' inside is plotted. Units are same as indicated on EISTAB product, with one exception: WAVG is multiplied by 100 for a more accurate plot.

UNITS : KNOTS, LVLS : THSD FT (MSL)

B R N = 10
 B+ = 790 (M/SEC)**2
 B- = -70 (M/SEC)**2
 SHR = 77 (M/SEC)**2
 WMAX = 38 M/SEC
 EL = 199 MB
 EL = 388 HND FT
 MPL = 452 HND FT

PARCEL FROM PMAX	
ENTRAINMENT = 60 PERCENT	
P0	= 963 MB
PMAX	= 870 MB
LCL	= 861 MB
LFC	= 718 MB
EI	= 20 J/KG X 10
EI+	= 27 (+ PART)
EI-	= -8 (- PART)
ENERGY CHANGE IN LAYERS	
P1	P2
870	718
718	672
672	662
662	313
313	100
	-7 J/KG X 10
	1
	0
	30
	-458

LI = 10
 KI = 37
 SWI = -3

CCL = 696 MB
 C TMP = 30 C
 C TMP = 85 F
 WAVG = 675 G/KG X 10-2

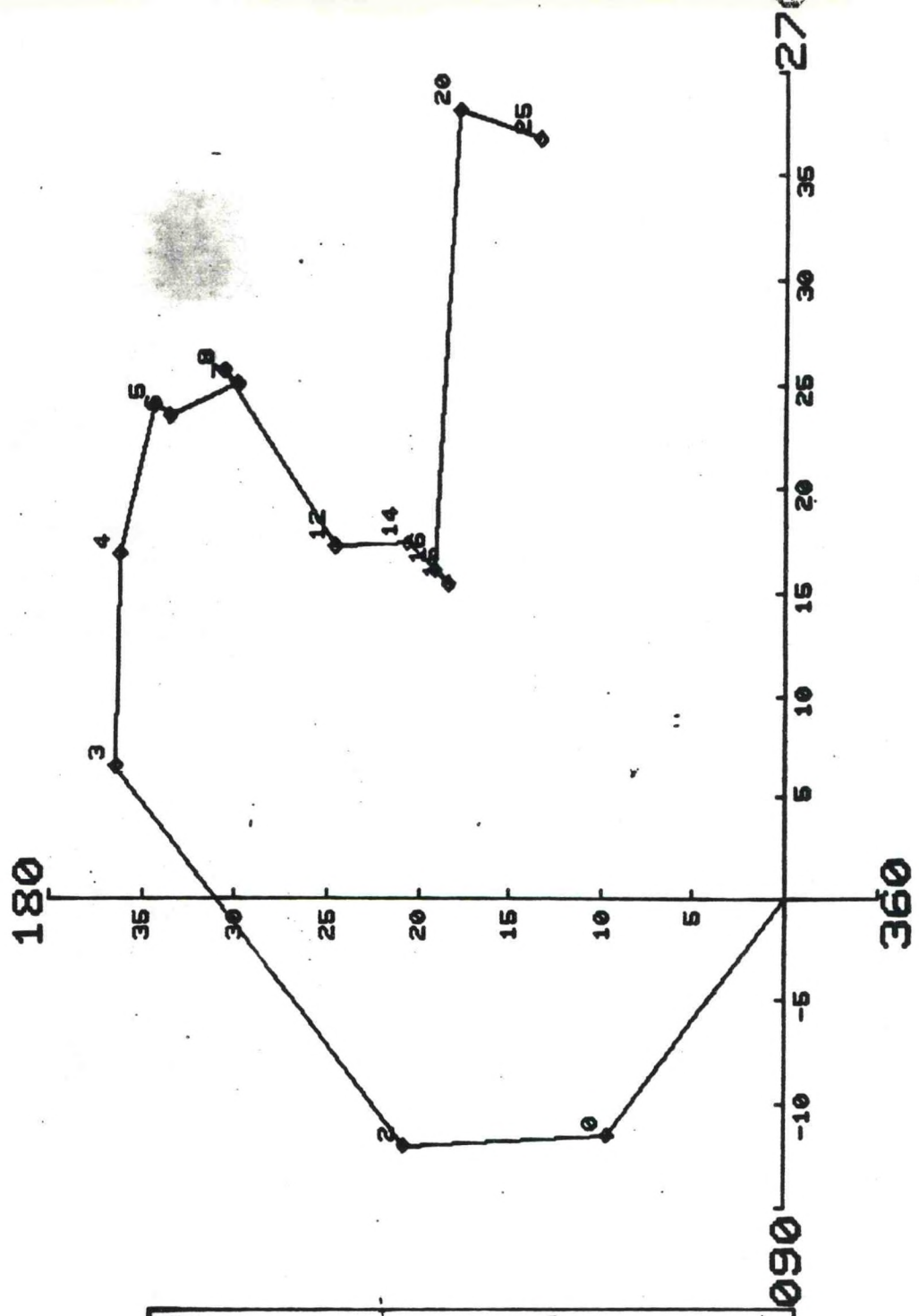


Fig. 6. Example of hodograph to 6 kilometers above ground level. Wind units are knots with data points labelled in thousands of feet MSL. Convective parameters on left enclosed in box are computed with parcel entrainment, but outside of box, no entrainment is used. Parameters that could not be computed are indicated by 'XXX'.

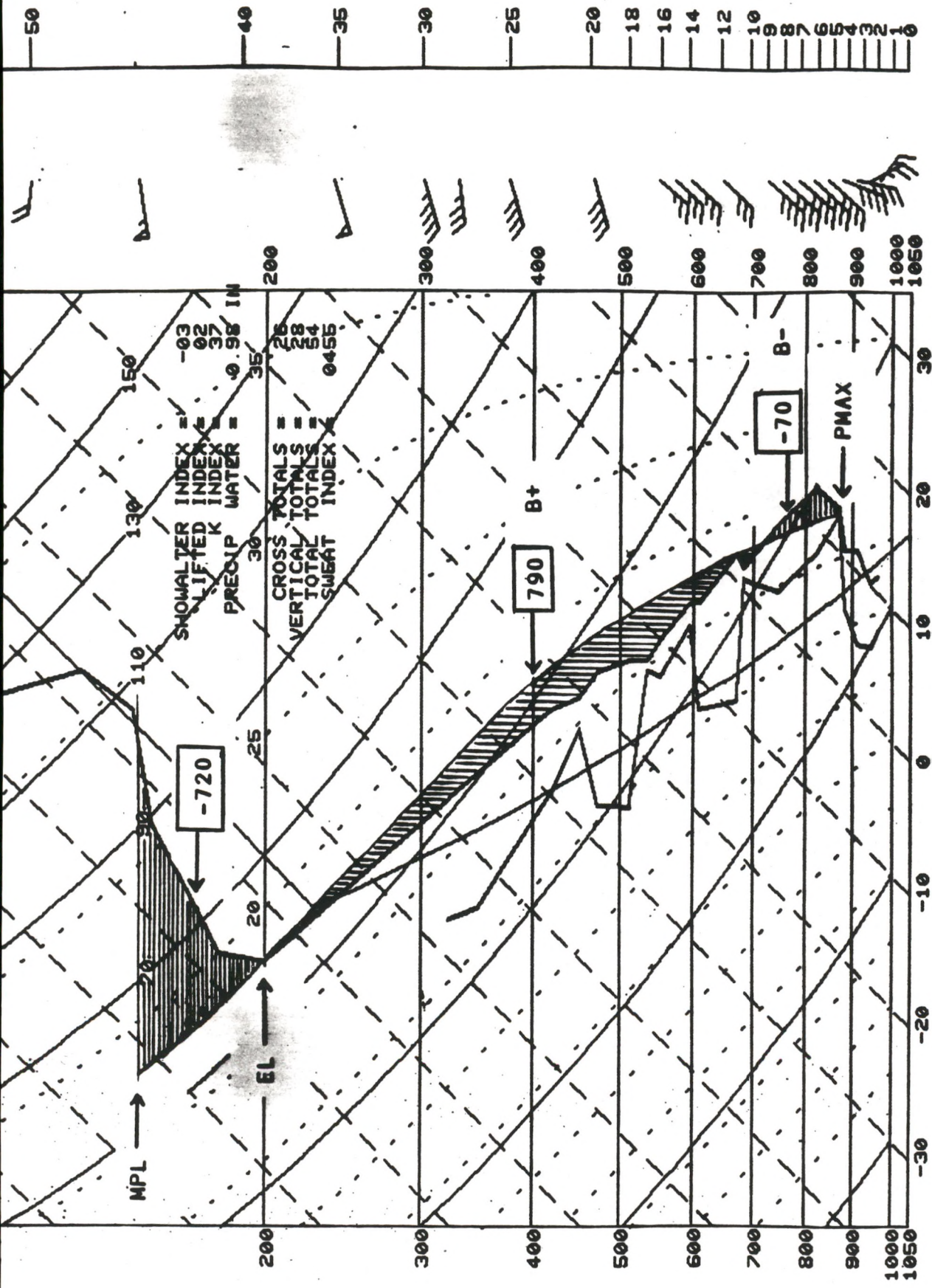
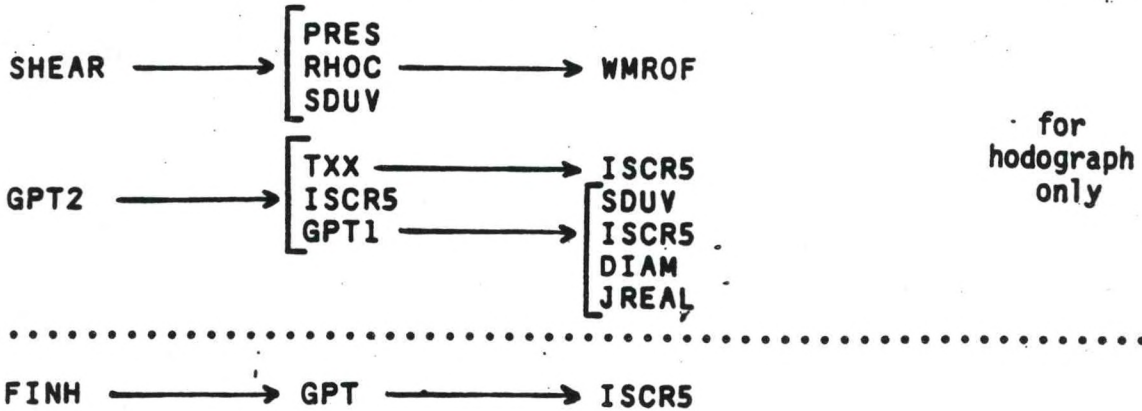
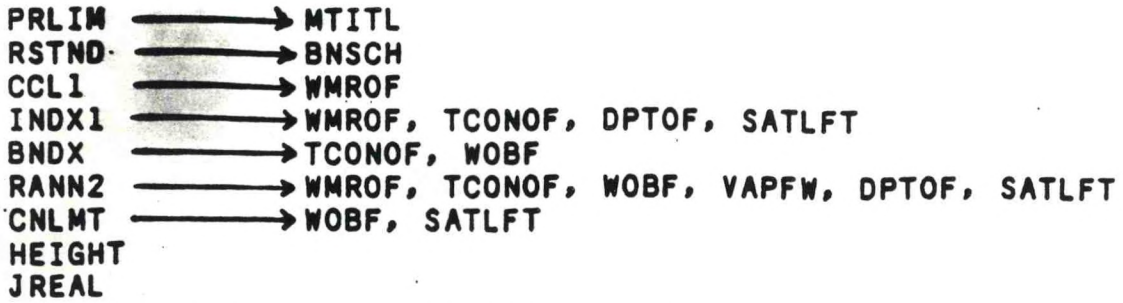


Fig. 7. Skew-T plot of the UMN raob, whose hodograph and convective parameters are shown in Fig. 6. Energy areas shown are for a non-entraining parcel rising from the level PMAX; these are the energy areas used for the BRN and MPL computations. values of the three energy areas are given by the numbers in the boxes with positive energy having vertical hatching and negative horizontal.

Main Program

CONVECT

Subroutines



Load Line

```

RLDR CONVECT RANN2 CCL1 INDX1 BNDX WOBF SATLFT TCONOF
WMROF DPTOF VAPFW HEIGHT JREAL ISCR5 GPT2 TXX
RSTND BNSCH MTITL GPT1 DIAM SDUV
[ PRLIM, SHEAR RHOC PRES CNLMT, FINH GPT ]
OUT TOP.LB AG.LB UTIL.LB FORT.LB

```

Fig.8. Software structure and load line for CONVECT program.

C PROGRAM CONVECT REV 01.00
 C APR 1986 STONE, H. M. ERH FTS 649-5443
 C FORTRAN IV/ REV 5.57 DG ECLIPSE (S230) RDOS/ REV 6.17
 C LOAD LINE: RLDR CONVECT RANN2 CCL1 INDX1 BNDX WDBF SATLFT TCONOF
 C WPROF DPTOF VAPFW HEIGHT JREAL ISCR5 GPT2 TX
 C RSTND BNSCH MTITL GPT1 DIAM SDUV
 C [PRLIM, SHEAR RHOC PRES CNLMT, FINH GPT]
 C OUT TOP.LB AG.LB UTIL.LB FORT.LB

C PURPOSE

C COMPUTES PARAMETERS USEFUL IN FORECASTING CONVECTION, MAPS
 C SELECTED PARAMETERS, AND DRAWS HODOGRAPHS.
 C INPUT IS FROM DATA FILES DECODED BY 'TTBBD' PROGRAM.
 C OUTPUT IS TO AFOS PRODUCTS: 'EISTAB', 'NMGCPHEIS',
 C 'NMGCPHEIT', AND HODOGRAPHS 'CCCHO' (CCC = 3 LETTER ID).
 C 'NMGCPHEIT' IS OPTIONAL AND CONTAINS DATA SPECIFIED IN A TWO
 C OR FOUR NUMERAL CODE SPECIFIED IN THE COMMAND LINE.

C EXTERNALS

C PRLIM OV0 OVERLAY
 C SHEAR OV1 OVERLAY
 C RHOC OV1 OVERLAY
 C PRES OV1 OVERLAY
 C CNLMT OV1 OVERLAY
 C RSTND
 C CCL1
 C RANN2
 C INDX1
 C BNDX
 C HEIGHT
 C GPT2
 C FINH OV2 OVERLAY
 C GPT OV2 OVERLAY

C CHANNELS/FILES

C MCHN - FOR OVERLAYS OV0, OV1, & OV2
 C JCHN - 'TTBBSTAS.DB', COMPLETE LIST OF RAOB STATIONS
 C ICHN - SIGNIFICANT LEVEL DATA FILE
 C KCHN - MANDATORY LEVEL DATA FILE
 C MK=1 - FORTRAN CHANNEL FOR 'INDEXX' FILE (EISTAB)

C EXITS

C STOP 121 - "RDS, JCHN, IER = "
 C UNABLE TO READ 'TTBBSTAS.DB'
 C
 C STOP 129 - "NOT IN STDIR"
 C STATION NOT PRESENT IN 'STDIR.MS'
 C
 C STOP 137 - "ERROR IN CONVECT"
 C SHOULD NEVER HAPPEN
 C
 C STOP 103 - "CMD LINE ERR, L =.. "(PRLIM)
 C COMMAND LINE ERROR
 C
 C STOP 138 - "RDS, LCHN, IER = " (PRLIM)
 C CANNOT READ 'STN' FILE
 C
 C STOP - "ERROR IN CCL1" (CCL1)
 C SHOULD NEVER HAPPEN
 C
 C STOP - "ERROR IN CNLMT" (CNLMT)
 C SHOULD NEVER HAPPEN

```

PARAMETER NRAOB=50 ; MAX NO. OF RAOB STATIONS
PARAMETER LV=31 ; MAX NO. OF LEVELS + 1 EXTRA LEVEL FM BNDX
PARAMETER MHOD=10 ; MAX NO. OF RAOBS FOR HODOGRAPH
PARAMETER NT=2 ; MAX NO. OF GRAPHIC PLOTS, EIS, EIT

```

.....

PARAMETERS APPEAR IN FOLLOWING SUBROUTINES:

```

NRAOB: FINH, GPT, GPT2
LV : GPT1, GPT2, CCL1, RANN2, CNLMT, SHEAR, RHOC, INDX1, BNDX
MHOD : PRLIM
NT : FINH, GPT

```

NAMED COMMON ALSO APPEARS IN FOLLOWING SUBROUTINES

```

S : CCL1, RANN2, CNLMT, INDX1, BNDX
T : RANN2
CCL : CCL1, RANN2, GPT2
G : RANN2, GPT2
TT : BNDX, GPT2
V : BNDX
C : PRLIM
WND : GPT1
PS : SHEAR, PRES, HEIGHT
ZZ : SHEAR, PRES, HEIGHT

```

.....

```

COMMON/S/KDATE(3), I HOUR, JNO, JJNO, P(0:LV), TS(0:LV), TSD(0:LV)
COMMON/T/RLCL, RLFC, EL, B2, B2P, B2N, IALL, B1, B1P, B1N, EX
COMMON/CCL/PCCL, ETCC, TS0, TSD0, L, TSCCL, TCCL, TDCCL, WAVG
COMMON/G/PP(0:20), ET(20), TJ(0:LV), DP, EFF, KMOD, KK
COMMON/TT/PT(0:LV), TST(0:LV), TSDT(0:LV)
COMMON/V/JNOM, PX
DIMENSION IXX(NRAOB), IYY(NRAOB), JR(NRAOB, NT), JL(NRAOB, NT),

```

```

1 IST(NRAOB, 2)

```

```

DIMENSION IBUF(100), IK(6)

```

IN COMMON/C, "X" IS NECESSARY TO PROTECT IT(1).

```

COMMON/C/MV(4), A(12), IDU(4), ID(2), ID3(3), X, IT(16), JHOD(MHOD, 2)

```

```

COMMON/WND/ZW(0:LV), DW(0:LV), SW(0:LV)

```

```

COMMON/PS/SP(10), SPP(0:10)

```

```

COMMON/ZZ/Z(10)

```

```

COMMON/ZZ/ZP(0:10)

```

```

EXTERNAL OV0, OV1, OV2

```

HGTS OF MANDATORY LEVELS (U.S. STANDARD ATMOS.)

```

DATA Z/111., 1457., 3012., 5574., 7185., 9164., 10363., 11784., 13608., 16180./

```

MANDATORY PRES LEVELS

```

DATA SP/1000., 850., 700., 500., 400., 300., 250., 200., 150., 100./

```

```

MK=1 ; FORTRAN CHANNEL NO. FOR "INDEXX" FILE

```

```

NBS=74 ; NUMBER OF BYTES PER STATION IN MANDATORY FILE

```

```

CNVM=.032808399 ; CONVERSION FACTOR, M TO FT X 10-2

```

```

DP=50. ; 50 MB STEP

```

```

IALL=1 ; IALL=2 TO PRINT EVERY LVL IN RANN2 SUBROUTINE

```

```

PX=400. ; CUT-OFF PRESSURE FOR ENERGY INDEX

```

```

N=0 ; COUNTER FOR NUMBER OF RAOB STATIONS

```

```

EFF0=60. ; NORMAL ENTRAINMENT RATE IN PERCENT

```

LOAD OVERLAY 0 FOR READING OPTIONAL PLOT VARIABLES & OPENING DATA FILES

```

CALL GCHN(MCHN, IER)

```

```

IF(IER.NE.1) TYPE "GCHN OV0, IER = ", IER

```

```

CALL OVOPN(MCHN, "CONVECT.OL", IER) ; OPEN CONVECT.OL

```

```

IF(IER.NE.1) TYPE "OVOPN OV0, IER = ", IER

```



```

CALL OVLOD(MCHN,OV0,-1,IER) ; LOAD OV0
IF(IER.NE.1) TYPE 'OVLOD OV0, IER = ',IER
CALL KLOSE(MCHN,IER)
IF(IER.NE.1) TYPE 'KLOSE OV0, IER = ',IER

```

```

CALL PR.LIM(MK,DP,EFF0,PX,ICHN,KCHN,NHOD,KDATE,IL)

```

```

OPEN FILE THAT HAS RAOB STN ID'S

```

```

CALL GCHN(JCHN,IER)
IF(IER.NE.1) TYPE 'JCHN, IER = ',IER
CALL OPENR(JCHN,'TTBBSTAS.DB',0,IER)
IF(IER.NE.1) TYPE 'OPENR, JCHN, IER = ',IER
IHOD=1 ; INDICATOR THAT HODOGRAPH TO BE DONE
JOD=0 ; COUNTER FOR HODOGRAPH RAOBS
JSTOP=0 ; INDICATOR THAT FINAL HODOGRAPH IS FINISHED

```

```

LOAD OVERLAY 1 FOR HODOGRAPHS

```

```

CALL GCHN(MCHN,IER)
IF(IER.NE.1) TYPE 'GCHN OV1, IER = ',IER
CALL OVOPN(MCHN,'CONVECT.OL',IER) ; OPEN CONVECT.OL
IF(IER.NE.1) TYPE 'OVOPN OV1, IER = ',IER
CALL OVLOD(MCHN,OV1,-1,IER) ; LOAD OV1
IF(IER.NE.1) TYPE 'OVLOD OV1, IER = ',IER
CALL KLOSE(MCHN,IER)
IF(IER.NE.1) TYPE 'KLOSE OV1, IER = ',IER

```

```

.....
START MAIN LOOP
.....

```

```

120 IRD=1 ; INDICATOR THAT SGL LEVELS READ OK
SHR=0. ; INDICATOR THAT 'SHEAR' MAY BE CALLED
CALL RDS(JCHN,IBUF,11,IER) ; READ STATION AFOS IDENTIFIERS
IF(IER.EQ.9) GO TO 8 ; END OF FILE
IF(IER.EQ.1) GO TO 121 ; NORMAL
TYPE 'RDS, JCHN, IER = ',IER
CALL FORKE('CONVECT','TTBBSTAS.DB',IER)
STOP 121
121 N=N+1 ; COUNT RAOBS
M=N
IF(JSTOP.EQ.1) GO TO 124 ; NO MORE HODOGRAPHS TO DO
IF(IHOD.EQ.0) GO TO 127 ; TEST FOR HODOGRAPH RAOB STATION
JOD=JOD+1
IHOD=0
127 IF(IBUF(4).NE.JHOD(JOD,1)) GO TO 124 ; TEST WHETHER TO DO HODO
IF(IBUF(5).NE.JHOD(JOD,2)) GO TO 124
IHOD=1 ; STATION ID FOUND
IF(JOD.EQ.NHOD) JSTOP=1
M=-M ; CHANGE TO NEGATIVE, TO INDICATE HODOGRAPH BEING COMPUTED
124 CONTINUE
CALL TYPED(M) ; TYPES + OR - N ON DASHER TO MONITOR PROGRESS, - HODO
KN=MOD(N,20)
IF(KN.EQ.0) WRITE(10,49) ; NEXT LINE
48 FORMAT(1H)
IST(N,1)=IBUF(4)
IST(N,2)=IBUF(5) ; STATION ID FOR PLOTTING
ID(1)=IBUF(4)

```

```

ID(2)=IBUF(5)
CALL UNPACK(ID,4,IDU) ; UNPACK STATION ID
IDU(4)=32 ; INSERT SPACE FOR BNSCH SUBROUTINE
CALL PACK(IDU,8,ID3)
ID3(3)=20040K ; DOUBLE SPACE

```

```

C .....
C READ SIG LEVEL DATA
CALL RDS (ICHN,IBUF,376,IER) ; READ STATION FM SIG DATA FILE
IF (IER.NE.1) TYPE 'RDS SIG, IER = ',IER
JNO1=IBUF(1) ; NO. OF SIG LVLS
ISYN=IBUF(5) ; SYNOPTIC STN NO. -999 IF MISG DATA
JNO=JNO1-1
IF (ISYN.NE.-999) GO TO 108 ; DATA AVBL
IRD=0
GO TO 126
108 IW=0
C SET INDEX
IK(1)=0
DO 105 I=2,6
105 IK(I)=IK(I-1)+IL ; INDEX FOR READING IBUF ARRAY (SGL FILE)
C
C READ SGL LEVELS AND WIND DATA
C
DO 102 I=1,JNO1
DO 107 J=1,6
107 IK(J)=IK(J)+1
II=I-1
P(II)=IBUF(IK(1)) ; PRESSURE
TS(II)=IBUF(IK(2))*0.1 ; TEMPERATURE
IF (IBUF(IK(3)).NE.-9999) GO TO 111
TSD(II)=TS(II)-30. ; IF DEWPOINT MISSING ASSUME DRY
GO TO 112
111 TSD(II)=IBUF(IK(3))*0.1 ; DEWPOINT
112 IF (IBUF(IK(4)).EQ.-9999) GO TO 102 ; WIND NOT AVBL
IW=IW+1 ; COUNT NO. OF WIND LEVELS
ZW(II)=IBUF(IK(4)) ; WIND LEVEL (HNDS OF FT)
DW(II)=IBUF(IK(5)) ; WIND DIRECTION (DEG)
SW(II)=IBUF(IK(6)) ; WIND SPEED (KTS)
102 CONTINUE
IF (JNO1.LT.IL) GO TO 114 ; READ REST OF WIND LVLS, IF ANY
GO TO 143
C
C READ WIND LEVELS ONLY
C
114 J1=JNO1+1
DO 113 I=J1,IL
DO 116 J=1,6
116 IK(J)=IK(J)+1
II=I-1
IF (IBUF(IK(4)).EQ.-9999) GO TO 113 ; WIND NOT AVBL
IW=IW+1 ; COUNT NO. OF WIND LVLS
ZW(II)=IBUF(IK(4)) ; WIND LEVEL
DW(II)=IBUF(IK(5)) ; WIND DIRECTION
SW(II)=IBUF(IK(6)) ; WIND SPEED
113 CONTINUE
C
C FOLLOWING CHECK NEEDED TO DETECT NEGATIVE PRESSURE FROM TT88D DECODER,
C A VERY RARE OCCURENCE.
143 IF(P(JNO).GT.0.) GO TO 126 ; NORMAL
WRITE (10,146) JNO,P(JNO)

```

```

146  FORMAT(1X,"BAD PRESSURE, P(",12," ) = ",F7.0)
      IRD=0 ;   CONSIDER ENTIRE RAOB AS MISSING
C
C .....
C  READ MAN DATA FILE
126  CALL RDS (KCHN,IBUF,NBS,IER) ;   READ ONE STATION FM MAN FILE
      IF (IER.NE.1) TYPE "RDS MAN, IER = ",IER
C  GET SFC ELEVATION & STN COORDS FM STDIR FILE
      CALL RSTND(ID3,ISYN0,ELV,ALAT,ALON,IX1,IY1,IX2,IY2,IX3,IY3,IZOOM,IER1)
      IF (IER1.EQ.1) GO TO 119 ;   STATION FOUND
      WRITE (10,129) (ID3(I),I=1,2)
129  FORMAT (1X,2A2," NOT IN STDIR")
      CALL FORKE("CONVECT","NOT IN STDIR",IER)
      STOP 129
119  IXX(N)=IX1+IX1 ;   X COORD
      IYY(N)=IY1+IY1 ;   Y COORD
      IF (IRD.EQ.0) GO TO 29 ;   SIG LEVEL DATA NOT AVAILABLE
      ZW(0)=ELV ;   SFC ELEVATION (METERS)
      IF (IW.LE.1) GO TO 144 ;   ONLY SFC WIND OR NO WIND
      IW1=IW-1
      DO 131 K=1,IW1
131  ZW(K)=ZW(K)*30.48 ;   CONVERT (FT/100) TO METERS
144  DO 132 K=1,10
      IF (IBUF(8+K).NE.-999) GO TO 133
      ZP(K)=Z(K) ;   STANDARD ATMOS HEIGHT VALUES
      GO TO 132
133  ZP(K)=IBUF(8+K) ;   HGTS FROM MANDATORY RAOB DATA
132  CONTINUE
      TROP=IBUF(19) ;   TROPOPAUSE PRESSURE
C
C  CHECK STN ELEVATION VS 1ST RAOB WIND LEVEL. IF OK, ADJUST PRESSURE/
C  HEIGHT ARRAYS FOR 'HEIGHT' AND 'PRES' SUBROUTINES.
      IF (IW.GT.1) GO TO 147 ;   AT LEAST 1 WIND LVL ABV SFC AVBL
      SHR=-1. ;   INDICATES WIND NOT AVBL TO 6KMS
      GO TO 138
147  IF (ZW(1).GT.ELV) GO TO 138 ;   NORMAL
C  FIRST WIND LEVEL ABOVE GROUND MAY BE FROM MANDATORY LEVEL OF RAOB AND
C  HEIGHT IS ROUNDED TO NEAREST 100 FEET. THIS MAY CAUSE FIRST WIND
C  LEVEL TO BE SLIGHTLY BELOW STATION ELEVATION. FOLLOWING STATEMENT
C  CORRECTS THIS SITUATION AND DOES NOT DAMAGE WIND SHEAR CALCULATION.
      ZW(1)=ELV
C  ADJUST ARRAYS FOR PRESSURE/HEIGHT RELATIONSHIP (DEPENDS ON SFC PRESSURE)
138  ZP(0)=ELV
      DO 136 I=1,10
      IF (P(0)-SP(I)) 136,136,137
136  CONTINUE
      TYPE "ERROR IN CONVECT"
      STOP 137 ;   SHOULD NEVER REACH THIS STATEMENT
137  IF (ZP(I).LE.ZP(I-1)) I=I+1 ;   TEST NECESSARY WHEN STANDARD ATMOS USED
      NS=11-I
      DO 139 J=1,NS
      ZP(J)=ZP(J+I-1)
139  SPP(J)=SP(J+I-1)
      SPP(0)=P(0) ;   SFC PRESSURE
C
C .....
C  ENERGY AREA COMPUTATIONS
C  KMOD=-1 ;   FOR ETCCCL CALCULATION
      CALL CCL1
C  CALL MODRB ;   MODIFIES RAOB FOR ETCCCL COMPUTATION
C  CALL RANN2 (PT,TST,TSMT,JNOM,PX) ;   CALLED FOR ETCCCL ONLY, JNOM FM MODRB

```

```

CALL INDX1 (RLI,RKI,RWI,$26,PTOP)
CALL BNDX (PTOP,$26) ; MODIFY RA0B FOR MAX INSTABILITY
EFF=0. ; ZERO ENTRAINMENT FOR EQUILIBRIUM LVL
KMOD=0 ; FOR NORMAL COMPUTATION WITH RANN2
CALL RANN2 (PT,TST,TSDD,JNOM,PX) ; COMPUTE EL LVL, JNOM FROM BNDX
CALL CALMT(B2,EL,TEL,TCTOP,CTOP,EA,RLCL) ; GET MAX PARCEL LEVEL

```

```

C
C FOLLOWING 7 LINES FOR TEST
C WRITE (10,201) (ID3(1),I=1,2),B2,EA,CTOP,EL,KK,RLCL,RLFC,P(JNO)
C201 FORMAT(1X,2A2,4F10.1,14,2X,3F10.1)
C KK1=KK-1
C DO 302 ITT=1,KK1
C JJJ=ITT-1
C302 WRITE (10,303) PP(JJJ),PP(ITT),ET(ITT)
C303 FORMAT (1X,3F9.2)
C

```

```

B2P0=B2P*10. ; FOR BULK RICHARDSON NO.
B2N0=B2N*10.
WMAX=0. ; INDICATES NO (NET) POSITIVE ENERGY AREA
IF(B2.GT.0.) WMAX=(2.*B2*10.)**.5 ; MAX VERTICAL VELOCITY
EL0=EL ; SAVE EL WITH ZERO ENTRAINMENT RATE
EFF=EFF0 ; RESET ENTRAINMENT RATE FOR ALL OTHER COMPUTATIONS
CALL RANN2 (PT,TST,TSDD,JNOM,PX) ; COMPUTE STABILITY INDICES
EL=EL0 ; USE EL WITH ZERO ENTRAINMENT
P0=P(0) ; FOR GPT2

```

```

C
C .....
C VARIABLES FOR OUTPUT TO 'EISTAB' (ALSO USED BY GPT2 FOR HODOGRAPH)

```

```

A(1)=P(0)-PT(0) ; P0-PMAX
A(2)=WMAX ; WMAX AT EL
A(3)=EL ; EL (MB)

```

```

C FOLLOWING 3 ARE IN MBS TO BE CONVERTED TO FEET LATER

```

```

A(4)=EL ; EL (FT)
A(5)=CTOP ; MPL(FT) MAXIMUM PARCEL LEVEL
A(6)=TROP ; TROP(FT)

```

```

C
A(7)=B1P ; + PART OF EI
A(8)=B1N ; - PART OF EI
A(9)=WAVG*100. ; AVG MIXING RATIO, LOWEST 100MB (G/KG X 10-2)
A(10)=RLI ; LIFTED INDEX
A(11)=RKI ; K INDEX
A(12)=RWI ; SHOWALTER INDEX

```

```

C
C A(4), A(5), A(6) IN MB (MISSING VALUES, NEGATIVE) ... CONVERT TO FEET

```

```

DO 14 I=4,6
IF(A(I).LT.0.) GO TO 14 ; VARIABLE IS MISSING
CALL HEIGHT(NS,A(I),A(I)) ; CONVERT MB TO METERS
A(I)=A(I)*CNVM ; CONVERT M TO FT X 10-2
14 CONTINUE

```

```

C
C STORE VARIABLE FOR PLOT
JR(N,1)=JREAL(B1) ; EI
JL(N,1)=JREAL(A(4)) ; EL (FT)
IF(MV(4).EQ.0) GO TO 16 ; NO OPTIONAL VARIABLE(S), SO NO EIT
JR(N,2)=JREAL(A(MV(4))) ; RIGHT SIDE VARIABLE
IF(MV(3).EQ.0) GO TO 17 ; LEFT SIDE VARIABLE NOT SPECIFIED
JL(N,2)=JREAL(A(MV(3))) ; LEFT SIDE VARIABLE
GO TO 16
17 JL(N,2)--999
16 CONTINUE

```

```

C
C OUTPUT FOLLOWS, FOR 'EISTAB'
  A(9)=A(9)*.01 ; CONVERT BACK TO G/KG
  IF (IBUF(1).NE.-999) GO TO 12 ; MANDATORY LEVELS USED
C MANDATORY LEVELS NOT USED TO CONVERT MB TO FT
  WRITE (MK,20) N,(ID3(1),I=1,2),(A(1),I=1,12),(ID3(1),I=1,2)
28  FORMAT ("<15><12>",I2,1X,2A2,F4.0,3F6.0,"E",F5.0,1X,F6.0,F5.0,
  1 F6.0,F6.2,3F5.0,1X,2A2)
  IF(IHOD.EQ.1) GO TO 140
  GO TO 120
C MANDATORY LEVELS USED TO CONVERT MB TO FT
12  WRITE (MK,22) N,(ID3(1),I=1,2),(A(1),I=1,12),(ID3(1),I=1,2)
22  FORMAT ("<15><12>",I2,1X,2A2,F4.0,3F6.0,1X,F5.0,1X,F6.0,F5.0,
  1 F6.0,F6.2,3F5.0,1X,2A2)
  IF(IHOD.EQ.1) GO TO 140
C
C NO HODOGRAPH
  GO TO 120 ; GO TO START OF LOOP TO READ NEXT STATION
C
C .....
C HODOGRAPH AND WIND SHEAR COMPUTATION
C
140 IF(JSTOP.EQ.1) IHOD=0 ; FINAL HODOGRAPH TO BE DONE
  IF(SHR.EQ.-1.) GO TO 140 ; WINDS NOT AVAILABLE TO 6KM
  CALL SHEAR(JNO1,P,TS,TSO,IW,ZW,DW,SW,SHR,I2,NS)
  BRICH=B2P0/SHR ; BULK RICHARDSON NO.
  IW=I2 ; NO. OF WIND LVLS TO 6KM, FOR HODOGRAPH
140 A(9)=A(9)*100. ; MOVING DECIMAL POINT OF 'WAVG' FOR GRAPHIC PLOT
  CALL GPT2(N,IRD,IST,RLCL,RLFC,A,B1,B2P0,B2N0,BRICH,SHR,P0,IT,IW)
C
C HODOGRAPH FINISHED, GO TO START OF LOOP FOR NEXT STATION
  GO TO 120
C
C ALL OR PART OF RAOB DATA MISSING BELOW
26  WRITE (MK,55) N,(ID3(1),I=1,2),PTOP
55  FORMAT ("<15><12>",I2,1X,2A2," SHORT RAOB",F7.0," MB")
  SHR=-2. ; INDICATOR TO DO HODOGRAPH ONLY, NO ALPHANUMERICS
  GO TO 57
29  WRITE (MK,56) N,(ID3(1),I=1,2)
56  FORMAT ("<15><12>",I2,1X,2A2," MISSING")
57  DO 30 I=1,NT ; SET ALL VARIABLE TO MISSING
  JR(N,I)=-999
  JL(N,I)=-999
30  IF(IHOD.EQ.1) GO TO 141
  GO TO 120
141 IF(JSTOP.EQ.1) IHOD=0 ; LAST HOD STATION FOUND, DATA MISSING
C
  CALL GPT2(N,IRD,IST,RLCL,RLFC,A,B1,B2P0,B2N0,BRICH,SHR,P0,IT,IW)
C
  GO TO 120 ; READ NEXT STATION
C
C CONTINUE
8  WRITE (10,48) ; GO TO NEXT LINE
C
C .....
C END OF MAIN LOOP
C
C .....
C CLOSE FILES
  CALL CLOSE (MK,IER)
  IF (IER.NE.1) TYPE "CHANNEL MK CLOSE ERROR, IER = ",IER
  CALL KLOSE (JCHN,IER)

```

```

IF (IER.NE.1) TYPE 'KLOSE JCHN, IER = ',IER
CALL KLOSE (KCHN,IER)
IF (IER.NE.1) TYPE 'KLOSE KCHN, IER = ',IER
CALL KLOSE (ICHN,IER)
IF (IER.NE.1) TYPE 'KLOSE ICHN, IER = ',IER

```

C
C

```

LOAD OVERLAY 2 FOR FINH -- FINISHES 'EISTAB' AND DOES 'EIS' & 'EIT'
CALL GCHN(MCHN,IER)
IF (IER.NE.1) TYPE 'GCHN OV2, IER = ',IER
CALL OVOPN(MCHN,'CONVECT.OL',IER) ; OPEN CONVECT.OL
IF (IER.NE.1) TYPE 'OVOPN OV2, IER = ',IER
CALL OVLOD(MCHN,OV2,-1,IER) ; LOAD OV2
IF (IER.NE.1) TYPE 'OVLOD OV2, IER = ',IER
CALL KLOSE(MCHN,IER)
IF (IER.NE.1) TYPE 'KLOSE OV2, IER = ',IER

```

C
C

```

FINISH EISTAB PRODUCT & DO GRAPHICS EIS, EIT
CALL FINH(N,IXX,IYY,JR,JL,IST,IT,MV)
STOP
END

```

*

*

```

OVERLAY OV0
PARAMETER MHOD=10 ; MAX NO. OF HODOGRAPHS
SUBROUTINE PRLIM (MK,DP,EFF0,PX,ICHN,KCHN,NHOD,KDATE,IL)

```

C
C

```

REV 01.00
MAR 1986 STONE, H. M. ERH FTS 649-5443
FORTRAN IV/ REV 5.57 DG ECLIPSE (S230) RDOS/ REV 6.17

```

C
C

```

PURPOSE
FIRST SUBROUTINE CALLED BY CONVECT. READS OPTIONAL NUMERIC
CODE SPECIFYING VARIABLE FOR GRAPHIC 'EIT'.
OPENS SGL AND MAN DATA FILES, WRITES TITLE FOR 'EISTAB', &
READS HODOGRAPH STATION FILE (STN).

```

C
C

```

ARGUMENT LIST
MK - FORTRAN CHANNEL NO. FOR 'INDEXX' = 'EISTAB'
DP - PRESSURE STEP FOR ENERGY AREA COMPUTATION
EFF0 - ENTRAINMENT RATE FOR EI COMPUTATION
PX - UPPER LIMIT OF ENERGY INTEGRATION FOR 'EI'
ICHN - CHANNEL NO. FOR SGL DATA FILE
KCHN - CHANNEL NO. FOR MAN DATA FILE
NHOD - NO. OF HODOGRAPH STATIONS LISTED IN 'STN'
KDATE - ARRAY WITH MON, DAY, YEAR OF DECODED DATA
IL - MAXIMUM NO. OF LEVELS DECODED BY 'TTBBD'

```

C
C

```

EXTERNALS
MTITL
CHANNELS/FILES

```

C
C

```

ICHN - FOR READING SGL DATA (ALSO FILNAM.TX)
KCHN - FOR READING MAN DATA
LCHN - FOR HODOGRAPH LIST 'STN'
MK - FORTRAN CHANNEL FOR 'INDEXX'

```

C
C

```

EXITS
STOP '103 CMD LINE ERR, L = ... , MV(3) = ... , MV(4) = ... '
ONLY 0, 2, OR 4 DIGITS, AFTER 'CONVECT' & MV.LE.12

```

C
C

```

STOP '130 RDS, LCHN, IER = ....'
CANNOT READ 'STN' FILE

```

C
C

```
COMMON/C,MV(4),A(12),IDU(4),ID(2),ID3(3),X,IT(16),JHOD(MHOD,2)
DIMENSION IBUF(100),KDATE(3),IFI(6),IFJ(6),ICUF(3)
```

```
C
MV(1)=4 ; EL (FT) , LEFT SIDE (EIS GRAPHIC)
MV(2)=7 ; EI INDEX, RIGHT SIDE (EIS GRAPHIC)
MV(3)=6 ; VARIABLE TO BE PLOTTED ON LEFT SIDE OF EIT GRAPHIC
MV(4)=6 ; VARIABLE TO BE PLOTTED ON RIGHT SIDE OF EIT GRAPHIC
```

```
C
CALL FCOM(IC, IER)
CALL COMCM(IC, IDU, L, ID, IER) ; READ. 'RUN:CONVECT'
IF (IER.NE.1) GO TO 100
CALL COMCM(IC, IDU, L, ID, IER) ; READ 2 OR 4 DIGIT NO.; VRBLS FOR PLOT
IF (IER.EQ.9) GO TO 101 ; NO OPTIONAL PLOT VARIABLES SPECIFIED
IF (IER.EQ.1.AND.(L.EQ.3.OR.L.EQ.5)) GO TO 103 ; 2 OR 4 DIGITS
```

```
C CALL FORKE, HERE, RESULTS IN FATAL RUNTIME ERROR 1
```

```
100 WRITE (10,126) L,MV(3),MV(4)
126 FORMAT (1X,"CMD LINE ERR, L = ",12," MV(3) = ",12," MV(4) = ",12)
STOP 103
```

```
103 CALL UNPACK(IDU,L,IT)
IF (L.EQ.5) GO TO 104 ; 4 DIGIT NO. READ
MV(4)=(IT(1)-48)*10+(IT(2)-48) ; SINGLE VARIABLE PLOTTED ON RIGHT
GO TO 102
```

```
104 MV(3)=(IT(1)-48)*10+(IT(2)-48)
MV(4)=(IT(3)-48)*10+(IT(4)-48)
```

```
102 CONTINUE
IF (MV(3).GT.12.OR.MV(4).GT.12) GO TO 100 ; ONLY 12 VARIABLES AVBL
CALL KLOSE(IC, IER)
IF (IER.NE.1) TYPE "KLOSE, SUBROUTINE PRLIM, IER = ",IER
```

```
C
101 CALL CFILW("HMSGPH.01",2,IER) ; CREATING GRAPHIC FILE FOR EIS & EIT
IF (IER.NE.1) WRITE (10,35) IER
```

```
35 FORMAT (1H,"IER = ",14," CFILW, PROGRAM CONVECT, STATEMENT 35")
```

```
C .....
C GET FILENAME FOR SGL DATA
```

```
CALL GCHN (ICHN, IER)
IF (IER.NE.1) TYPE "GCHN, IER = ",IER
CALL OPENR (ICHN,"FILNAM.TX",0,IER)
IF (IER.NE.1) TYPE "OPENR, IER = ",IER
CALL RDS (ICHN,IFI,12,IER)
IF (IER.NE.1) TYPE "RDS, IER = ",IER
CALL KLOSE (ICHN, IER)
IF (IER.NE.1) TYPE "KLOSE FILNAM.TX, IER = ",IER
```

```
C
C GET FILENAME FOR MAN DATA
DO 125 I=1,6
```

```
125 IFJ(I)=IFI(I)
IFJ(4)="S."
```

```
C
C .....
C OPEN CHANNEL AND READ 1ST SIXTEEN WORDS OF MAN FILE
```

```
CALL GCHN (KCHN, IER)
IF (IER.NE.1) TYPE "GCHN KCHN, IER = ",IER
CALL OPENR (KCHN,IFJ,0,IER) ; MAN LEVEL FILE
IF (IER.NE.1) TYPE "OPENR KCHN, IER = ",IER
CALL RDS (KCHN,IBUF,32,IER) ; READ 1ST 16 WORDS
```

```
C
C READ 1ST SIXTEEN WORDS OF SGL DATA FILE
```

```
CALL GCHN (ICHN, IER)
IF (IER.NE.1) TYPE "GCHN, IER = ",IER
CALL OPENR (ICHN,IFI,0,IER) ; SIG LEVEL FILE
```

```

IF (IER.NE.1) TYPE 'OPENR, IER = ', IER
CALL RDS (ICHN, IBUF, 32, IER) ; READ 1ST 16 WORDS
IF (IER.NE.1) TYPE 'RDS, IER = ', IER
KDATE(1)=IBUF(7) ; MONTH
KDATE(2)=IBUF(6) ; DAY
KDATE(3)=IBUF(8)-1900 ; YEAR, 2 DIGIT
IHOUR=IBUF(5) ; HOUR
IL=IBUF(9) ; MAX NO. OF LVLS DECODED

C
CALL MTITL (IHOUR, KDATE, IT) ; CREATE DATE/TIME GROUP FOR TITLE
DO 90 I=1, 14
90 IT(I)=IT(I+2) ; SHIFT TITLE FOR 2 DIGIT TIME
C
C .....
C WRITE TITLE FOR 'EISTAB' PRODUCT
CALL FOPEN (MK, 'INDEXX', 300)
WRITE (MK, 21) (KDATE(1), I=1, 3), IHOUR, DP, EFF0, PX
21 FORMAT (12X, ' RAOB PARAMETERS FOR ', I2, '/', I2, '/', I2,
1 2X, I2, 'Z', 3X, 'DP = ', F4.0, 3X, 'EFF = ', F5.0, 3X, 'PX = ', F5.0)
WRITE (MK, 22)
22 FORMAT ('<15><12>', 'UNITS: (MB) (M/SEC) (MB) (...FT X 100...)' ,
1 '(J/KG X 10)(G/KG) (...DEG C....)')
WRITE (MK, 23)
23 FORMAT ('<15><12>', 3X, 'STN', 1X, 'P0MX', 2X, 'WMAX', 3X, 'EL', 4X,
1 'EL', 4X, 'MPL', 2X, 'TROP', 3X, 'EI+', 3X, 'EI-', 2X, 'WAVG', 2X, 'LI', 3X,
2 'KI', 2X, 'SWI', 2X, 'STN')
WRITE (MK, 27)
27 FORMAT ('<15><12>', 8X, '(1)', 3X, '(2)', 3X, '(3)', 3X, '(4)', 3X, '(5)', 3X,
1 '(6)', 3X, '(7)', 3X, '(8)', 2X, '(9)', 2X, '(10)', 1X, '(11)', 1X, '(12)')
C
C .....
C OPEN FILE FOR STATIONS WITH HODOGRAPH
CALL GCHN(LCHN, IER)
IF (IER.NE.1) TYPE 'LCHN, IER = ', IER
CALL OPENR(LCHN, 'STN', 0, IER)
IF (IER.NE.1) TYPE 'OPENR, LCHN, IER = ', IER
DO 136 I=1, NHOD ; COUNT RAOBS IN 'STN' AND FILL JHOD ARRAY
CALL RDS(LCHN, ICUF, 5, IER)
IF (IER.NE.1) GO TO 138 ; ERROR
IF (ICUF(1).EQ.'00') GO TO 137
JHOD(I, 1)=ICUF(1)
136 JHOD(I, 2)=ICUF(2)
GO TO 137
138 TYPE 'RDS, LCHN, IER = ', IER
CALL FORKE('CONVECT', 'BAD HODO LIST', IER)
STOP 138
137 CALL KLOSE(LCHN, IER)
IF (IER.NE.1) TYPE 'KLOSE LCHN, IER = ', IER
NHOD=I-1
WRITE (10, 106) (IFI(I), I=1, 6), NHOD ; SGL FILE NAME & NO. OF HODOS
106 FORMAT (1X, 6A2, 4X, 'HODOGRAPHS = ', I2)
RETURN
END

```

*

*

SUBROUTINE MTITL (IHOUR, KDATE, IT)

C REV 01.01

C MAR 1985

STONE, H. M.

ERH FTS 649-5443

PURPOSE

MAKES DATE/TIME GROUP TITLE FOR A GRAPHIC.
 CONVERTS HOUR, DATE, AND YEAR TO ASCII CHARACTERS AND MAKES
 3 LETTER MONTH ABBREVIATION.
 BEGINNING AND ENDING HOUR OUTPUT TO THE MINUTE.

ARGUMENT LIST

Ihour - INPUT: HOUR (4 DIGITS)
 KDATE - INPUT: MON,DATE,YEAR
 IT - OUTPUT: DATE/TIME GROUP TITLE FOR PLOT

DIMENSION KDATE(3),IT(16),ISC(7)

GET ASCII TIME

CALL ISCR5(ISC,Ihour,+1)

DO 15 I=1,4

K=I+2

IF (ISC(K).EQ.32) ISC(K)=48 ; LEADING ZEROES IN TIME RESTORED

IT(I)=ISC(K)

IT(5)=90 ; Z

IT(6)=32 ; SPACE

GET ASCII DATE

CALL ISCR5 (ISC,KDATE(2),+1)

IT(7)=ISC(5)

IT(8)=ISC(6)

IT(9)=32 ; SPACE

IT(13)=32 ; SPACE

GET ASCII YEAR

CALL ISCR5 (ISC,KDATE(3),+1)

IT(14)=ISC(5)

IT(15)=ISC(6)

IT(16)=0 ; MUST BE ZERO FOR TEXT SUBROUTINE

GET 3 LETTER MONTH ABBREVIATION

KGO=KDATE(1)

GO TO (1,2,3,4,5,6,7,8,9,10,11,12) KGO

IT(10)=74 ; JAN

IT(11)=65

IT(12)=78

GO TO 14

IT(10)=70 ; FEB

IT(11)=69

IT(12)=66

GO TO 14

IT(10)=77 ; MAR

IT(11)=65

IT(12)=82

GO TO 14

IT(10)=65 ; APR

IT(11)=88

IT(12)=82

GO TO 14

IT(10)=77 ; MAY

IT(11)=65

IT(12)=89

GO TO 14

IT(10)=74 ; JUN

IT(11)=85

IT(12)=78

GO TO 14

```

7   IT(10)=74   ;   JUL
    IT(11)=85
    IT(12)=76
    GO TO 14-
8   IT(10)=65   ;   AUG
    IT(11)=85
    IT(12)=71
    GO TO 14
9   IT(10)=83   ;   SEP
    IT(11)=69
    IT(12)=80
    GO TO 14
10  IT(10)=79   ;   OCT
    IT(11)=67
    IT(12)=84
    GO TO 14
11  IT(10)=78   ;   NOV
    IT(11)=79
    IT(12)=86
    GO TO 14
12  IT(10)=68   ;   DEC
    IT(11)=69
    IT(12)=67
14  CONTINUE
    RETURN
    END

```

*

*

SUBROUTINE RSTND (ISTN, ISYNO, ELV, ALAT, ALON, IX1, IY1, IX2, IY2, IX3, IY3, IZOOM, IER1)

```

C           REV 01.00
C   JUN 1984           STONE, H. M.           ERH   FTS 649-5443
C   FORTRAN IV/ REV 5.20  DG ECLIPSE (S230)   RDOS/REV 7.20
C   PURPOSE
C   GIVEN STATION ID, RETURNS ALL INFORMATION FOR THAT STATION
C   FROM THE STATION DIRECTORY FILE (STDIR.MS)
C   ARGUMENT LIST
C   ISTN           - STATION ID
C                   OUTPUT:
C   ISYNO          - SYNOPTIC NO.
C   ELV            - ELEVATION (METERS)
C   ALAT           - LATITUDE (HUNDREDTH OF DEGREES)
C   ALON           - LONGITUDE (HUNDREDTH OF DEGREE)
C   IX1, IY1       - COORDINATES FOR NA MAP BACKGROUND
C   IX2, IY2       - COORDINATES FOR US MAP BACKGROUND
C   IX3, IY3       - COORDINATES FOR ROTATED NA MAP BACKGROUND
C   IZOOM          - ZOOM THRESHOLD
C   IER1           - 1 IF ISTN FOUND, 0 IF NOT FOUND
C   EXTERNALS
C   BNSCH

```

```

DIMENSION ISTN(3), IAD(2), IB(3), IC1(14), IC2(14), IC3(14)
IFLDP=1
IFLD=6
IAD(1)=0
IAD(2)=0
CALL GCHN(ICHN, IER)
IF (IER.NE.1) TYPE "GCHN ERROR - RSTND, IER = ", IER
CALL OPENR(ICHN, "STDIR.MS", 0, IER)

```

```

IF (IER.NE.1) TYPE 'OPENR ERROR - RSTND, IER = ',IER
CALL RDS (ICHN,IB,6,IER) ; READ 1ST 3 WORDS FROM FILE
IF (IER.NE.1) TYPE 'RDS ERROR - RSTND, IER = ',IER
CALL BNSCH (ICHN,IB(1),IB(2),IB(3),IFLDP,IFLD,ISTN,IAD,IC1,IC2,IC3,IC)
IF (IC.EQ.0) GO TO 5
GO TO (1,2,3),IC
1 ISYNO=IC1(4)
  ELV=IC1(5)
  ALAT=IC1(6)*.01 ; SET DECIMAL POINT
  ALON=IC1(7)*.01
  IX1=IC1(8)
  IY1=IC1(9)
  IX2=IC1(10)
  IY2=IC1(11)
  IX3=IC1(12)
  IY3=IC1(13)
  IZOOM=IC1(14)
  IER1=1
  GO TO 4
2 ISYNO=IC2(4)
  ELV=IC2(5)
  ALAT=IC2(6)*.01
  ALON=IC2(7)*.01
  IX1=IC2(8)
  IY1=IC2(9)
  IX2=IC2(10)
  IY2=IC2(11)
  IX3=IC2(12)
  IY3=IC2(13)
  IZOOM=IC2(14)
  IER1=1
  GO TO 4
3 ISYNO=IC3(4)
  ELV=IC3(5)
  ALAT=IC3(6)*.01
  ALON=IC3(7)*.01
  IX1=IC3(8)
  IY1=IC3(9)
  IX2=IC3(10)
  IY2=IC3(11)
  IX3=IC3(12)
  IY3=IC3(13)
  IZOOM=IC3(14)
  IER1=1
  GO TO 4
5 IER1=0
4 CONTINUE
  CALL KLOSE (ICHN,IER)
  IF (IER.NE.1) TYPE 'KLOSE ERROR - RSTND, IER = ',IER
  RETURN
  END

```

*

*

```

SUBROUTINE BNSCH(ICHN,NREC,LREC,ISTAR,IFLDP,IFLD,ITEST,
1 IAD,IC1,IC2,IC3,IC)
C BINARY SEARCH ROUTINE:
C
C PROGRAMMER - RICH THOMAS SXB,ISL,SDO 11/79

```

C
C
C
C
C
C
C
C
C
C
C

```
ICHN=CHANNEL WHICH FILE HAS BEEN OPENNED TO
NREC=NUMBER OF RECORDS
LREC=LENGTH OF EACH RECORD (BYTES)
ISTAR=BYTE OF FIRST RECORD (0-BEGINNING)
IFLDP=WORD POINTER TO FIELD IN RECORD
IFLD=LENGTH OF FIELD IN BYTES
ITEST=ARRAY CONTAINING TEST FIELD
IAD=RETURNED TWO WORD ARRAY CONTAINING ADDRESS ITEST RECORD
SHOULD BEGIN AT-
  IC= 1,2,3 IN SECOND WORD INDICATING RECORD WAS FOUND AND
  IS IN ARRAY IC1,IC2, OR IC3
THOSE THREE ARRAYS SHOULD BE DIMENSIONED LREC/2 WORDS
DIMENSION ITEST(1),IC1(1),IC2(1),IC3(1),IAD(2)
DIMENSION IAD1(2),IAD2(2),IAD3(2)
DIMENSION D1(2),D2(2)
INTEGER D1,D2
IC=0
IAD1(1)=0
IAD1(2)=ISTAR
CALL SPOS(ICHN,IAD1,IER)
CALL ERROR(IER,'I1')
CALL RDS(ICHN,IC1,LREC,IER)
CALL ERROR(IER,'RDS - IC1')
D2(1)=0
D2(2)=LREC
CALL DSUB(D2,D2,IAD1)
CALL DMPY(D1,NREC,LREC)
CALL DSUB(IAD2,D1,D2)
CALL SPOS(ICHN,IAD2,IER)
CALL ERROR(IER,'I2')
CALL RDS(ICHN,IC2,LREC,IER)
CALL ERROR(IER,'RDS-IC2')
CALL BCOMP(IC1(IFLDP),ITEST,IFLD,IER1)
IF(IER1.GT.1)GO TO 100
CALL BCOMP(IC2(IFLDP),ITEST,IFLD,IER2)
IF(IER2.NE.2)GO TO 125
5 CALL DSUB(D1,IAD2,IAD1)
CALL DDVD(INC,IR,D1,LREC)
IF(INC.GE.32767)GO TO 900
IF(INC.LT.1)GO TO 150
INC=(INC-1)/2+1
CALL DMPY(D1,INC,LREC)
CALL DADD(IAD3,IAD1,D1)
CALL SPOS(ICHN,IAD3,IER)
CALL ERROR(IER,'I5')
CALL RDS(ICHN,IC3,LREC,IER)
CALL ERROR(IER,'I6')
CALL BCOMP(IC3(IFLDP),ITEST,IFLD,IER3)
IF(IER3.EQ.1)GO TO 50
IF(IER3.EQ.2)GO TO 60
IF(IER3.NE.3)GO TO 900
IAD(1)=IAD3(1)
IAD(2)=IAD3(2)
IC=3
RETURN
50 IAD1(1)=IAD3(1)
   IAD1(2)=IAD3(2)
   GO TO 5
60 IAD2(1)=IAD3(1)
```

```

IAD2(2)=IAD3(2)
IF(INC.EQ.1)GO TO 150
GO TO 5
100 IAD(1)=IAD1(1)
IAD(2)=IAD1(2)
IF(IER1.NE.3)GO TO 101
IC=1
IAD(1)=IAD1(1)
IAD(2)=IAD1(2)
101 RETURN
125 D1(1)=0
D1(2)=LREC
CALL DADD(IAD,D1,IAD2)
IF(IER2.NE.3)GO TO 126
IAD(1)=IAD2(1)
IAD(2)=IAD2(2)
IC=2
126 RETURN
150 IAD(1)=IAD3(1)
IAD(2)=IAD3(2)
RETURN
900 CALL ERROR(IER3,'IER3')
IER=2
CALL ERROR(IER,'TOO MANY RECORDS IN FILE')
STOP
END

```

*

*

```

PARAMETER LV=31
SUBROUTINE CCL1

```

```

C          REV 01.10
C          MAR 1986          STONE, H. M.          ERH   FTS 649-5443
C          FORTRAN IV/ REV 5.57  DG ECLIPSE (S230)  RDOS/ REV 6.17
C          PURPOSE
C          COMPUTES 'WAVG' AVERAGE MIXING RATIO IN LOWEST 100 MBS.
C          CCL LEVEL, AND CONVECTIVE TEMPERATURE.
C          IF WAVG INTERSECTS RA08 TEMPERATURE LINE MORE THAN ONCE, THE
C          UPPER INTERSECTION IS TAKEN FOR THE CCL.
C          ARGUMENT LIST
C          NONE          - ALL VARIABLES PASSED IN NAMED COMMON
C          VARIABLES
C          WAVG          - MEAN MIXING RATIO, LOWEST 100MB (G/KG)
C          PCCL          - PRESSURE AT CCL (MB)
C          TCCL          - TEMPERATURE AT CCL (DEG C)
C          TDCCL        - DEWPOINT AT CCL (DEG C)
C          TSCCL        - CONVECTIVE TEMPERATURE (DEG C) AT SURFACE
C          EXITS
C          STOP          ERROR IN CCL1
C                   SHOULD NEVER HAPPEN
C
COMMON/S/KDATE(3), I HOUR, JNO, JJNO, P(0:LV), TS(0:LV), TSD(0:LV)
COMMON/CCL/PCCL,ETCCL,TS0,TS00,L,TSCCL,TCCL,TDCCL,WAVG
THETA(T,P2,P1)=T*(P2/P1)**.2857142 ; DRY ADIABATIC (T,P1) TO (THETA,P2)
DP1=100. ; AVERAGES MIXING RATIO OVER FIRST -DP1- MBS.
WSUM=0.
J=0
P1=P(0)
TDE1=TSD(0)

```

```

W1=WMROF(P(0),TDE1)
PFINISH=P1-DP1
3 P2=PFINISH
IF (P(J+1)-P2) 1,2,2
2 P2=P(J+1)
TDE2=TS(J+1) ; ENVIRONMENT DEWPT AT P2
J=J+1
GO TO 9
1 PLOG1=ALOG(P(J)/P(J+1))
FACTOR=(TS(J)-TS(J+1))/PLOG1
PLOG2=ALOG(P2/P(J+1))
TDE2=TS(J+1)+FACTOR*PLOG2 ; ENVIRONMENT DEWPT AT P2
9 W2=WMROF(P2,TDE2) ; MIXING RATIO AT P2
PLOG3=ALOG(P1/P2)
W=.5*(W1+W2)*PLOG3 ; AVG MIX RATIO IN Lyr P1-P2
WSUM=WSUM+W
P1=P2
W1=W2
IF (P2.GT.PFINISH) GO TO 3
C COMPUTE AVG VALUES FOR FIRST -DP1- MBS.
PLOG4=ALOG(P(0)/PFINISH)
WAVG=WSUM/PLOG4
C DETERMINE LAYER CONTAINING CCL, CHECKING FROM TOP OF ATMOS DOWNWARD
DO 4 I=0,JNO
II=JNO-I
WS=WMROF(P(II),TS(II))
IF (WS-WAVG) 4,5,6
4 CONTINUE
8 WRITE (10,0)
FORMAT (1H," ERROR IN CCL1")
STOP
5 PCCL=P(II)
TCCL=TS(II)
TDCCL=TS(II)
TSCCL=THETA(TCCL+273.16,P(0),PCCL)-273.16 ; CONVECTIVE TEMP DEG C
L=JNO+1
JJNO=JNO-1
RETURN ; CCL LEVEL IS ALSO A RAOB SIGNIFICANT LEVEL
6 J=II+1
JJNO=JNO
IF (J.EQ.(JNO+1)) GO TO 32 ; MIXING RATIO AT TOP OF RAOB > WAVG
C MXG RATIO INTERSECTS ENVIRONMENTAL TEMP BTWN P(J) AND P(J-1)
C THIS LAYER WILL BE SUBDIVIDED UNTIL SATURATION VAPOR PRESSURE AT
C MIDPOINT OF LAYER IS SUFFICIENTLY CLOSE (.01 G/KG) TO WAVG.
C THIS DETERMINES THE CCL LEVEL.
P1=P(J-1) ; BOTTOM
P2=P(J) ; TOP
T1=TS(J-1) ; BOTTOM
T2=TS(J) ; TOP
31 ALOG1=ALOG(P1/P2)
PM=.5*(P1+P2) ; MIDPOINT PRESSURE
ALOG2=ALOG(PM/P2)
TPM=T2+(T1-T2)/ALOG1*ALOG2 ; MIDPOINT TEMPERATURE
WSM=WMROF(PM,TPM) ; MIDPOINT SATURATION MIXING RATIO
IF (ABS(WSM-WAVG).LE..01) GO TO 29 ; TEST FOR TOLERANCE
IF (WSM-WAVG) 28,29,30
28 P2=PM
T2=TPM
GO TO 31
30 P1=PM

```

```

T1=TPM
GO TO 31
29 PCCL=PM ; CCL PRESSURE
TCCL=TPM ; CCL TEMPERATURE
C COMPUTE DEWPOINT AT CCL LEVEL
ALOG1=ALOG(P(J-1)/P(J))
ALOG2=ALOG(PM/P(J))
TDCCL=TS(J)+(TS(J-1)-TS(J))/ALOG1*ALOG2
IF (TDCCL.GT.TCCL) TDCCL=TCCL ; CORRECTION FOR DEWPOINT EXCEEDING
1 TEMPERATURE BY SMALL AMT
TSCCL=THETA(TCCL+273.16,P(0),PCCL)-273.16 ; CONVECTIVE TEMP DEG C
L=J ; INDEX NUMBER OF ADDED CCL LEVEL, IF 'MODRB' CALLED
RETURN
C CCL NOT COMPUTED DUE TO EXCESSIVE MIXING RATIO AT TOP OF RAOB;
C THIS USUALLY HAPPENS WITH ABORTED RAOB, OR VERY UNUSUAL RAOB STRUCTURE
C SOMETIMES OBSERVED IN WINTER.
32 PCCL=-999. ; MISSING
RETURN
END

```

*

*

```

PARAMETER LV=31
SUBROUTINE INDX1 (RLI,RKI,RWI,Q,PTOP)
REV 01.10
C APR 1986 STONE, H. M. ERH FTS 649-5443
C FORTRAN IV/ REV 5.57 DG ECLIPSE (S230) RDOS/ REV 6.17
C PURPOSE
C COMPUTES LIFTED INDEX, K INDEX, AND SHOWALTER INDEX
C ARGUMENT LIST
C RLI - LIFTED INDEX
C RKI - K INDEX , = -999 IF SFC PRES < 850MB
C RWI - SHOWALTER INDEX, = -999 IF SFC PRES < 850MB
C Q - ABNORMAL RETURN STATEMENT NO.
C PTOP - PRESSURE AT TOP OF RAOB, FOR ABNORMAL RETURN
C
C IF SFC PRES LESS THAN 850MB, K AND SHOWALTER SET = -999.
COMMON/S/KDATE(3),IHOUR,JNO,JJNO,P(0:LV),TS(0:LV),TSD(0:LV)
DIMENSION PL(3),TL(3),TDL(3)
INTEGER Q
THETA(T,P2,P1)=T*(P2/P1)**.2857142 ; DRY ADIABATIC (T,P1) TO (THETA,P2)
IHI=0 ; INDICATOR FOR SFC PRESSURE GREATER THAN 850MB.
PL(1)=850.
PL(2)=700.
PL(3)=500.
DP1=50. ; AVERAGES OVER FIRST -DP1- MBS.
WSUM=0.
THSUM=0.
J=0
P1=P(0)
TE1=TS(0)
TDE1=TSD(0)
TH1=THETA(TE1+273.16,1000.,P(0)) ; POT TEMP
W1=WPROF(P(0),TDE1)
PFINISH=P1-DP1
3 P2=PFINISH
IF (P(J+1)-P2) 1,2,2
2 P2=P(J+1)
J=J+1

```

```

1  PLOG1=ALOG(P(J)/P(J+1))
   FACTORT=(TS(J)-TS(J+1))/PLOG1
   FACTORD=(TSD(J)-TSD(J+1))/PLOG1
   PLOG2=ALOG(P2/P(J+1))
   TE2=TS(J+1)+FACTORT*PLOG2 ; ENVIRONMENT TEMP AT P2
   TDE2=TSD(J+1)+FACTORD*PLOG2 ; ENVIRONMENT DEWPT AT P2
   TH2=THETA(TE2+273.16,1000.,P2) ; POT TEMP AT TE2,P2
   W2=WPROF(P2,TDE2) ; MIXING RATIO AT P2
   PLOG3=ALOG(P1/P2)
   TH=.5*(TH1+TH2)*PLOG3 ; AVG POT TEMP IN LYR P1-P2
   W=.5*(W1+W2)*PLOG3 ; AVG MIX RATIO IN LYR P1-P2
   THSUM=THSUM+TH
   WSUM=WSUM+W
   P1=P2
   TH1=TH2
   W1=W2
   IF (P2.GT.PFINISH) GO TO 3
C  COMPUTE AVG VALUES FOR FIRST -DP1- MBS.
   PLOG4=ALOG(P(0)/PFINISH)
   THAVG=THSUM/PLOG4
   WAVG=WSUM/PLOG4
   PPARCEL=P(0)-.5*DP1
   TPARCEL=THETA(THAVG,PPARCEL,1000.)-273.16 ; DEG C
   X=.0200*(TPARCEL-12.5+7500./PPARCEL) ; NON-IDEAL GAS CORRECTION
   WFW=1.+0.000045*PPARCEL+.00140*XXX ; NON-IDEAL GAS CORRECTION
   E2=.001*WAVG*PPARCEL/((WAVG*.001+.62197)*WFW) ; VAPOR PRES (MB)
   TDPARCEL=DPTOF(E2)
   TC=TCNOF(TPARCEL,TPARCEL)
   TH=THAVG-273.16 ; POT TEMP DEG C
   WTH=WDBF(TH)
   WTC=WDBF(TC)
   THW=TH-WTH+WTC ; EQUIV WET BULB POT TEMP (DEG C)
   TP500=SATLFT(THW,500.)
C  GET TEMP AND DEWPT AT 850,700,500 MBS
   DO 5 J=1,3
   DO 4 I=0,JNO
   IF (PL(J)-P(I)) 4,6,7
4  CONTINUE
   PTOP=P(JNO)
   RETURN Q
6  TL(J)=TS(I)
   TDL(J)=TSD(I)
   GO TO 5
7  IF (J.NE.1) GO TO 8
   IF (I.NE.0) GO TO 8
   IHI=1
   GO TO 5 ; SFC PRESSURE LESS THAN 850MB
8  FACTOR=ALOG(PL(J)/P(I))/ALOG(P(I-1)/P(I))
   TL(J)=TS(I)+FACTOR*(TS(I-1)-TS(I))
   TDL(J)=TSD(I)+FACTOR*(TSD(I-1)-TSD(I))
5  CONTINUE
   RLI=TL(3)-TP500 ; LIFTED INDEX
   IF (IHI.EQ.0) GO TO 9 ; COMPUTE K AND SHOWALTER INDICES
   RKI=-999. ; K INDEX MISG
   RWI=-999. ; SHOWALTER INDEX MISG
   RETURN
9  RKI=(TL(1)-TL(3))+TDL(1)-(TL(2)-TDL(2)) ; K INDEX
C  COMPUTE SHOWALTER INDEX
   TC=TCNOF(TL(1),TDL(1))
   TH=THETA(TL(1)+273.16,1000.,850.)-273.16 ; DEG C

```



```

WTH=W0BF(TH)
WTC=W0BF(TC)
THW=TH-WTH+WTC ; EQUIV WET BULB POT TEMP
TP=SATLFT(THW,500.)
RWI=TL(3)-TP ; SHOWALTER INDEX
RETURN
END

```

*

*

```

PARAMETER LV=31
SUBROUTINE BNDX (PTOP,Q)

```

```

C          REV 01.10
C MAR 1986          STONE, H. M.          ERH   FTS 649-5443
C FORTRAN IV/ REV 5.57  DG ECLIPSE (S230)  RDOS/ REV 6.17
C PURPOSE

```

```

C DETERMINES LEVEL OF MAXIMUM INSTABILITY IN THE LOWER 150MBS
C OF RAOB. ADJUSTS ORIGINAL RAOB, SO LEVEL OF MAX INSTABILITY
C IS FIRST SIGNIFICANT LEVEL AND ADDS ADDITIONAL PRESSURE
C LEVEL PX. IF PX IS NOT A SGFNT LEVEL. IF RAOB TERMINATES
C BELOW PX, IT IS EXTRAPOLATED TO PX. IF TOP LEVEL IS WITHIN
C 50 MBS OF PX.

```

ARGUMENT LIST

```

C PTOP          - PRESSURE AT TOP OF RAOB, ABNORMAL RETURN
C Q            - STATEMENT NO. FOR ABNORMAL RETURN

```

```

COMMON/S/KDATE(3), I HOUR, JNO, JJNO, P(0:LV), TS(0:LV), TSD(0:LV)

```

```

COMMON/TT/PT(0:LV), TST(0:LV), TSDT(0:LV)

```

```

COMMON/V/JNOM, PX

```

```

DIMENSION PB(2), TB(2), TDB(2)

```

```

INTEGER Q

```

```

THETA(T,P2,P1)=T*(P2/P1)**.2857142 ; DRY ADIABATIC (T,P1) TO (THETA,P2)
DP2=150.

```

```

C GET TEMP AND DEWPT AT P(0)-DP2 AND PX

```

```

PB(1)=P(0)-DP2

```

```

PB(2)=PX

```

```

DO 5 J=1,2

```

```

DO 4 I=0, JNO

```

```

IF (PB(J)-P(I)) 4,6,7

```

```

4 CONTINUE

```

```

I=I-1

```

```

IF (J.EQ.2.AND.(P(JNO)-PB(2)).LT.50.) GO TO 7 ; EXTRAPOLATES, IF WITHIN 50MBS

```

```

PTOP=P(JNO) ; RAOB TERMINATES TOO LOW

```

```

RETURN Q

```

```

6 TB(J)=TS(I)

```

```

TDB(J)=TSD(I)

```

```

GO TO 5

```

```

7 FACTOR=ALOG(PB(J)/P(I))/ALOG(P(I-1)/P(I))

```

```

TB(J)=TS(I)+FACTOR*(TS(I-1)-TS(I))

```

```

TDB(J)=TSD(I)+FACTOR*(TSD(I-1)-TSD(I))

```

```

5 CONTINUE

```

```

C FIND LARGEST POTENTIAL WET BULB TEMPERATURE IN FIRST DP2 MBS

```

```

THWMAX=-1000.

```

```

II=0

```

```

DO 1 I=0, JNO

```

```

IF (P(I)-PB(1)) 8,10,10

```

```

10 TC=TCNOF (TS(I),TSD(I))

```

```

TH=THETA(TS(I)+273.16,1000.,P(I))-273.16 ; DEG C

```

```

WTH=W0BF(TH)

```

```

WTC=W0BF(TC)
THW=TH-WTH+WTC ; WET BULB POTENTIAL TEMPERATURE
IF (THW-THWMAX) 1,1,2
2 THWMAX=THW
II=I
PMAX=P(I)
TMAX=TS(I)
TDMAX=TSD(I)
1 CONTINUE
8 IF (P(I-1).EQ.PB(1)) GO TO 9
TC=TCONOF (TB(1),TDB(1))
TH=THETA(TB(1)+273.16,1000.,PB(1))-273.16 ; DEG C
WTH=W0BF(TH)
WTC=W0BF(TC)
THW=TH-WTH+WTC ; WET BULB POT TEMP
IF (THW-THWMAX) 9,9,12
12 THWMAX=THW
II=I-1
PMAX=PB(1)
TMAX=TB(1)
TDMAX=TDB(1)
9 CONTINUE
C MODIFY RAOB SO LOWEST LEVEL HAS MAXIMUM WET BULB POTENTIAL TEMPERATURE
PT(0)=PMAX
TST(0)=TMAX
TSDT(0)=TDMAX
JNOO=JNO-II
DO 11 J=1,JNOO
PT(J)=P(J+II)
TST(J)=TS(J+II)
TSDT(J)=TSD(J+II)
11 CONTINUE
DO 14 J=1,JNOO
IF (PB(2)-PT(J)) 14,17,16
14 CONTINUE
J=JNOO+1
GO TO 20 ; EXTRAPOLATE RAOB
16 I=JNOO
18 TST(I+1)=TST(I) ; MOVE ALL LEVELS ABOVE PB(2) UP 1 LEVEL
TSDT(I+1)=TSDT(I)
PT(I+1)=PT(I)
I=I-1
IF (I.GE.J) GO TO 18 ; J SET IN DO 14 LOOP
20 TST(J)=TB(2) ; ADD TB(2) LEVEL
TSDT(J)=TDB(2)
PT(J)=PB(2)
JNOM=JNOO+1
GO TO 19
17 JNOM=JNOO
19 CONTINUE
RETURN
END

```

*

*

PARAMETER LV=31
SUBROUTINE RANN2 (PA, TSA, TSDA, JNOJ, PX)

C REV 01.00

C MAR 1986

STONE, H. M.

ERH FTS 649-5443

PURPOSE

COMPUTES ENERGY AREAS ON A THERMODYNAMIC DIAGRAM USING PARCEL_METHOD WITH SELECTED ENTRAINMENT RATE AND PRESSURE STEP. PARCEL LIFTED IS AT LOWEST LEVEL OF RAOB GIVEN IN ARGUMENT OF SUBROUTINE. TO COMPUTE ENERGY INDEX EI=B1, SUBROUTINE BNDX MUST BE CALLED FIRST TO MODIFY ORIGINAL RAOB. TO COMPUTE ETCL, SUBROUTINE MODRB MUST BE CALLED FIRST.

ARGUMENT LIST

- PA - PRESSURE LEVELS OF RAOB, PARCEL TO BE LIFTED FROM THE LOWEST PRESSURE LEVEL.
- TSA - TEMPERATURES AT 'PA' LEVELS
- TSDA - DEWPOINTS AT 'PA' LEVELS
- JNOJ - NO. OF LEVELS IN RAOB; PA, TSA, TSDA
- PX - PRESSURE LEVEL FOR ENDING THE EI COMPUTATION

VARIABLES

- KMOD - MUST BE SET = 0, FOR STANDARD COMPUTATION
MUST BE SET = -1, FOR ETCL COMPUTATION
- EFF - ENTRAINMENT RATE; PERCENT OF INCREASE IN MASS PER 500MB ASCENT
- KK - NO. OF ENERGY AREAS IN SOUNDING + 1
- PP - ARRAY OF PRESSURES (MB) WHERE PARCEL TEMPERATURE = SOUNDING TEMPERATURE, E.G. LIMITS OF + AND - ENERGY AREAS
- ET - ARRAY CONTAINING + AND - ENERGIES. ALL ENERGIES ARE J/KG, BUT HAVE BEEN DIVIDED BY 10.
- B1 - ENERGY INDEX EI (J/KG X 10)
- B1P - + PART OF B1
- B1N - - PART OF B1
- B2 - OLD ENERGY INDEX EI2, WHICH ENDED AT THE EQUILIBRIUM LEVEL
- B2P - + PART OF B2
- B2N - - PART OF B2

```

COMMON/S/KDATE(3), I HOUR, JNO, JJNO, P(0:LV), TS(0:LV), TSD(0:LV)
COMMON/G/PP(0:20), ET(20), TW(0:LV), DP, EFF, KMOD, KK
COMMON/T/RLCL, RLFC, EL, B2, B2P, B2N, IALL, B1, B1P, B1N, EX
COMMON/CCL/PCCL, ETCL, TS0, TSD0, L, TSCCL, TCCL, TDCCL, WAVG
DIMENSION PA(0:LV), TSA(0:LV), TSDA(0:LV)
THETA(T, P2, P1) = T * (P2/P1)**.2857142 ; DRY ADIABATIC (T, P1) TO (THETA, P2)
KDP=0 ; KDP RESET TO 1, IF 2ND PASS THRU RANN2, WITH REDUCED DP
DPSAVE=DP ; SAVES ORIGINAL DP, PASSED THRU COMMON/G
R=287.04 ; GAS CONSTANT FOR DRY AIR .. J/KG PER DEG K
R=R*.1 ; SCALING ENERGY UNITS
EF1=.00002*EFF ; ENTRAINMENT FACTOR PER MILLIBAR
100 IF (KMOD.EQ.-1) GO TO 106
    TS0=TSA(0)
    TSD0=TSDA(0)
106 IF (TS0.NE.TSD0) GO TO 92
    TC=TS0 ; PARCEL INITIALLY SATURATED
    RLCL=PA(0)
    GO TO 107
92 TC=TCONOF(TS0, TSD0) ; CONDENSATION TEMP
    PC=PA(0)*((TC+273.16)/(TS0+273.16))**.2857142 ; COND. PRES.
    IF (KMOD.EQ.-1) PC=PCCL ; PC COMPUTED ABOVE IS NOT EXACTLY PCCL
107 TH=THETA(TS0+273.16, 1000., PA(0))-273.16 ; POT. TMP DEG C
    WTH=W0BF(TH)
    WTC=W0BF(TC)
    THW=TH-WTH+WTC ; EQUIV WET BULB POT TMP (DEG C)
C LIFT DRY ADIABATICALLY UNTIL TP=TC AT PRESSURE PC
    
```

```

7 DO 7 I=1,20
  ET(I)=0.
  DT1=0.
  IF (KMOD.EQ.-1) DT1=TS0-TSA(0)
  J=0
  JJ=0
  JK=0
  KJ=0
  EN=0.
  EP=0.
  P1=PA(J)
  PP(0)=PA(0)
  KK=1
  KKK=0
  TP=TSA(J)
  IF (KMOD.EQ.-1) TP=TS0
  IF (TSDA(J).EQ.999.) TSDA(J)=TSA(J)-30. ; IF MISG. ASSUME DRY
  WP=WP/ROF(P1,TSDA(J))
  IF (IALLEQ.2) WRITE (10,86)
86 FORMAT (1H , "P1",8X, "P2",10X, "TE",13X, "TP",13X, "DT1",12X, "DT2",
1 12X, "E")
  IF (TS0.EQ.TSD0) GO TO 15 ; PARCEL INITIALLY SATURATED
13 P2=P1-DP
  MJ=0
  IF (PC-P2) 3,4,4
4 P2=PC
  RLCL=PC ; LIFTING CONDENSATION LVL
3 IF (PA(J+1)-P2) 5,6,6
6 P2=PA(J+1)
  J=J+1
  MJ=1
  KJ=1
30 PLOG1=ALOG(PA(J)/PA(J+1))
  FACTORT=(TSA(J)-TSA(J+1))/PLOG1
  IF (TSDA(J+1).EQ.999.) TSDA(J+1)=TSA(J+1)-30. ; IF MISG. ASSUME DRY
  FACTORD=(TSDA(J)-TSDA(J+1))/PLOG1
  KJ=1
5 IF (KJ.EQ.0) GO TO 30 ; INSURES FACTORT,D COMPUTED 1ST TIME THRU
  IF (JJ.EQ.0) TP0=TP ; SAVE ORIGINAL TP
  IF (JJ.EQ.1) TP=TP0 ; RESETS TP TO ORIGINAL VALUE, IF P2 ADJUSTED
  TP=TP+273.16 ; CONVERT TO DEG K
  TP=THETA (TP,P2,P1)-273.16 ; DRY ADIABATIC LIFT P1 TO P2 DEG C
  PLOG2=ALOG(P2/PA(J+1))
  TE=TSA(J+1)+PLOG2*FACTORT ; ENVIRONMENTAL TEMP AT P2
  DP1=P1-P2
  IF (KMOD.EQ.-1) GO TO 42 ; NO ENTRAINMENT BELOW CCL LEVEL
  IF (EFF.EQ.0.) GO TO 42 ; EFF=0. FOR NO ENTRAINMENT
  EF=EF1*DP1
  TP=(TP+273.16+EF*(TE+273.16))/(1.+EF)-273.16 ; DEG C
  TDE=TSDA(J+1)+PLOG2*FACTORD ; DEG C
  WE=WP/ROF(P2,TDE) ; G/KG MIXING RATIO OF ENVIRONMENT
  WP=(WP+EF*WE)/(1.+EF) ; MIXING RATIO OF PARCEL AFTER MIXING
  X=.0200*(TP-12.5+7500./P2) ; CORRECTION FOR NON-IDEAL GAS
  WFW=1.+0.0000045*P2+.00140*XXX ; CORRECTION FOR NON-IDEAL GAS
  E2=WP*.001*P2/((WP*.001+.62197)*WFW) ; VAPOR PRES (MB) OF PARCEL
  ES2=VAPFW(TP) ; SATURATION VAPOR PRES OF PARCEL
  ES=ES2-E2
  IF (ES) 40,40,41 ; GOES TO 40, IF PARCEL SATURATED
41 IF (ES.LE..01) GO TO 40 ; CLOSE ENOUGH FOR SATURATION
  TDP=DPTOF(E2) ; DEWPOINT OF PARCEL AFT MXG (DEG C)

```

```

TC=TCNOF (TP,TDP)
PC=P2*(TC+273.16)/(TP+273.16)**(1./2857142)
GO TO 42
40 TC=TP
PC=P2
RLCL=PC
C SINCE LCL HAS BEEN CHANGED, NEW -THW- IS ALSO REQUIRED
TH=THETA(TC+273.16,1000.,PC)-273.16 ; POT TEMP DEG C
WTH=W0BF(TH)
WTC=W0BF(TC)
THW=TH-WTH+WTC
42 DT2=TP-TE
IF (JJ.EQ.0) GO TO 96
TI=DT1-DT2
JJ=0
JK=1
IF (TI) 10,10,11
96 IF (KMOD.NE.-1) GO TO 14
IF (P2.GT.PC) GO TO 14
KKK=1
GO TO 11 ; LAST STEP IN COMPUTING CCL ENERGY
14 IF (JK.NE.1) GO TO 66 ; JK=1, IF PREVIOUS PASS WAS A CROSSING PT
JK=0
IF (DT2) 10,10,11
66 IF (DT2) 8,8,9
8 IF (DT1) 10,10,12
9 IF (DT1) 12,11,11
C GOES TO 12 IF DRY ADIABAT CROSSES ENVIRONMENTAL TEMP
12 P2=P1-ABS(DT1)/(ABS(DT1)+ABS(DT2))*DP1 ; APPROX PRES WHERE DT2=0.
IF (KK.LE.20) GO TO 75
TYPE "ET DIMENSION EXCEEDS 20"
GO TO 110
75 KKK=1
JJ=1
IF (MJ.EQ.0) GO TO 5
C MJ=1 MEANS J, WHICH HAS JUST BEEN SET AT STATEMENT 6, MUST BE RESET
C TO INTERPOLATE PROPERLY
J=J-1
MJ=0
GO TO 30
10 E=.5*(DT2+DT1)*ALOG(P1/P2)
EN=EN+E
IF (IALL.EQ.2) WRITE (10,85) P1,P2,TE,TP,DT1,DT2,E
85 FORMAT (1H ,2F10.3,5E15.6)
P1=P2
DT1=DT2
IF (KKK.EQ.0.AND.P2.NE.PA(JNOJ)) GO TO 62
ET(KK)=EN*R ; CONVERTS TO J/KG UNITS
PP(KK)=P2
KK=KK+1
EN=0.
KKK=0
62 IF (P2.EQ.PC) GO TO 15 ; PARCEL SATURATED
GO TO 13
11 E=.5*(DT2+DT1)*ALOG(P1/P2)
EP=EP+E
IF (IALL.EQ.2) WRITE (10,85) P1,P2,TE,TP,DT1,DT2,E
P1=P2
DT1=DT2
IF (KKK.EQ.0.AND.P2.NE.PA(JNOJ)) GO TO 63

```

```

ET(KK)=EP*R      ;   CONVERTS TO J/KG UNITS
PP(KK)=P2
KK=KK+1
EP=0.
KKK=0
63   IF (P2.EQ.PC) GO TO 15 ;   PARCEL SATURATED
      GO TO 13
C
C   LIFT PARCEL ALONG SATURATION ADIABATIC
C
15   CONTINUE
      IF (KMOD.NE.-1) GO TO 84
      DT1=0.
      ETCL=ET(1)
      RETURN      ;   REMOVE, IF FULL COMPUTATION OF CCL MODIFIED SOUNDING IS DESIRED
      KK=1      ;   KK SET FROM 2 BACK TO 1, CCL ENERGY HAS JUST BEEN COMPUTED
84   JJ=0
      JK=0
      ISTOP=0
      KKK=0
24   P2=P1-DP
      MJ=0
      IF (PA(J+1)-P2) 16,17,17
17   P2=PA(J+1)
      IF (PA(J+1).GT.PA(JNOJ)) GO TO 25
      ISTOP=1
      GO TO 16
25   J=J+1
      MJ=1
88   PLOG1=ALOG(PA(J)/PA(J+1))
      FACTORT=(TSA(J)-TSA(J+1))/PLOG1
      IF (TSDA(J+1).EQ.999.) TSDA(J+1)=TSA(J+1)-30. ;   IF MISG, ASSUME DRY
      FACTORD=(TSDA(J)-TSDA(J+1))/PLOG1
      KJ=1
16   IF (KJ.EQ.0) GO TO 88
      IF (JJ.EQ.0) THW0=THW      ;   SAVE ORIGINAL THW
      IF (JJ.EQ.1) THW=THW0 ;   RESETS THW TO ORIGINAL VALUE, IF P2 ADJUSTED
      TP=SATLFT (THW,P2) ;   TEMP OF PARCEL AT P2 ON -THW- WET ADIABAT
      PLOG2=ALOG(P2/PA(J+1))
      TE=TSA(J+1)+PLOG2*FACTORT ;   ENVIRONMENTAL TEMP AT P2
      DP1=P1-P2
      IF (EFF.EQ.0.) GO TO 67 ;   EFF=0. FOR NO ENTRAINMENT
      TDE=TSDA(J+1)+PLOG2*FACTORD ;   ENVIRONMENTAL DEWPT AT P2
      WE=WPROF (P2,TDE) ;   MIXING RATIO (G/KG) OF ENVIRONMENT
      WP=WPROF (P2,TP) ;   MIXING RATIO OF SATURATED PARCEL
      EF=EF1*DP1
      WP=(WP+EF*WE)/(1.+EF)
      TP=(TP+273.16+EF*(TE+273.16))/(1.+EF)-273.16
      X=.0200*(TP-12.5+7500./P2) ;   CORRECTION FOR NON-IDEAL GAS
      WFW=1.+0.0000045*P2+.00140*XXX ;   CORRECTION FOR NON-IDEAL GAS
      E2=WP*.001*P2/((WP*.001+.62197)*WFW) ;   VAPOR PRES (MB) OF PARCEL
      TDP=DPTOF(E2) ;   DEWPT OF PARCEL AFT MXG
      IF (TDP.GT.TP) TDP=TP
      TC=TCONOF(TP,TDP)
      TH=THETA(TP+273.16,1000.,P2)-273.16 ;   POT TEMP DEG C
      WTH=W0BF(TH)
      WTC=W0BF(TC)
      THW=TH-WTH+WTC ;   EQUIV WET BULB POT TEMP (DEG C)
      TP=SATLFT(THW,P2) ;   PARCEL TEMP AFT EVAPORATING LIQUID WATER
67   DT2=TP-TE

```

```

C IF ADDITIONAL INFORMATION ON LEVELS IS NEEDED, INSERT PRINT STATEMENT HERE
  IF (JJ.EQ.0) GO TO 23 ; JJ=1 IF NEW P2 HAS BEEN COMPUTED FOR CROSSOVER.
  T1=DT1-DT2
  JJ=0
  JK=1
  IF (T1) 20,20,22
23 IF (JK.NE.1) GO TO 65 ; JK=1, IF PREVIOUS PASS WAS A CROSSING PT
  JK=0

C
C IN CASE SAT. ADIABAT INTERSECTS ENVIRONMENTAL TEMP IN 2 PLACES CREATING
C A VERY SMALL POSITIVE AREA, THIS AREA WILL BE IGNORED (STATEMENT 101).
  CHECK=DT2*ET(KK-1) ; USUALLY NEGATIVE
  IF (CHECK.LT.0.) GO TO 100
  IF (DT2) 101,102,102
101 EN=ET(KK-1)
  KK=KK-1
C TYPE "STATEMENT 101 USED IN RANN2"
  GO TO 20

C IF STATEMENT 102 IS USED, PRESSURE STEP IS REDUCED AND ENTIRE COMPUTATION
C IS REPEATED. THIS OCCURS WHEN DT2 CHANGES SIGN SEVERAL TIMES IN A SHORT
C PRESSURE DISTANCE. THIS SHOULD BE A VERY RARE OCCURENCE
102 DP=10. ; REDUCE PRESSURE STEP TO 10MB.
  IF (KDP.EQ.0) GO TO 109
  GO TO 110 ; RANN2 CANNOT BE COMPLETED WITH REDUCED PRESSURE STEP
109 KDP=KDP+1
C TYPE "STATEMENT 102 USED IN RANN2"
  GO TO 100 ; REPEAT ENTIRE ENERGY CALCULATION WITH 10MB PRES STEP

C
100 IF (DT2) 20,20,22
65 IF (DT2) 10,10,19
10 IF (DT1) 20,20,21
19 IF (DT1) 21,22,22
C GOES TO 21 IF WET ADIABAT CROSSES ENVIRONMENTAL TEMP
21 P2=P1-ABS(DT1)/(ABS(DT1)+ABS(DT2))*DP1
  IF (KK.LE.20) GO TO 76
  TYPE "ET DIMENSION EXCEEDS 20"
  GO TO 110
76 KKK=1
  JJ=1
  IF (MJ.EQ.0) GO TO 16
C MJ=1 MEANS J, WHICH HAS JUST BEEN SET AT STATEMENT 25, MUST BE RESET
C TO INTERPOLATE PROPERLY
  J=J-1
  MJ=0
  GO TO 88
20 E=.5*(DT2+DT1)*ALOG(P1/P2)
  EN=EN+E
  IF (IALL.EQ.2) WRITE (10,85) P1,P2,TE,TP,DT1,DT2,E
  P1=P2
  DT1=DT2
  IF (P2.NE.PX) GO TO 99
  EX=EN*R ; SUBTOTAL FOR ENERGY AREA ENDING AT PX
  KX=KK
99 IF (KKK.EQ.0.AND.P2.NE.PA(JNOJ)) GO TO 60
  ET(KK)=EN*R ; CONVERTING TO J/KG UNITS
  PP(KK)=P2
  KK=KK+1
  EN=0.
  KKK=0
60 IF (ISTOP.EQ.1.AND.P1.EQ.PA(JNOJ)) GO TO 26

```

```

GO TO 24
22 E=.5*(DT2+DT1)*ALOG(P1/P2)
EP=EP+E
IF (IALL.EQ.2) WRITE (10,95) P1,P2,TE,TP,DT1,DT2,E
P1=P2
DT1=DT2
IF (P2.NE.PX) GO TO 184
EX=EP*R ; SUBTOTAL FOR ENERGY AREA ENDING AT PX
KX=KK
184 IF (KKK.EQ.0.AND.P2.NE.PA(JNOJ)) GO TO 61
ET(KK)=EP*R ; CONVERTING TO J/KG UNITS
PP(KK)=P2
KK=KK+1
EP=0.
KKK=0
61 IF (ISTOP.EQ.1.AND.P1.EQ.PA(JNOJ)) GO TO 26
GO TO 24
26 CONTINUE
C
C KK = NUMBER OF ENERGY AREAS IN SOUNDING + 1
KK1=KK-1
KK2=KK-2
KK3=KK-3
EL=-999.
B2=-999.
B2P=-999.
B2N=-999. ; -999 DENOTES THAT VARIABLE IS UNDEFINED
RLFC=0.
C DETERMINE -LFC- LEVEL
IF (KK.EQ.2.AND.ET(1).GT.0.) RLFC=PP(0)
IF (KK.EQ.3.AND.ET(1).GT.0.) RLFC=PP(0)
IF (KK.EQ.3.AND.ET(1).LT.0.) RLFC=PP(1)
IF (KK.GE.4.AND.ET(1).LT.0.) RLFC=PP(1)
IF (KK.GE.4.AND.ET(1).GT.0.) RLFC=PP(2)
C IN ALL OTHER CASES RLFC IS UNDEFINED...RLFC=0.
C
IF (ET(KK1).GT.0.) GO TO 70 ; HIGHEST AREA IS +, NO INDICES COMPUTED
C -EL- LEVEL DETERMINED HERE
EL=PP(KK2)
C COMPUTE ENERGY INDICES BELOW EL LEVEL
IF (KK.EQ.2) GO TO 70 ; ONLY ONE LAYER, ALL NEGATIVE
B2=0.
B2P=0.
B2N=0.
IF (ET(1).LT.0.) GO TO 58
DO 74 I=1,KK2,2
74 B2P=B2P+ET(I)
IF (KK.GT.3) GO TO 68
B2N=0.
GO TO 69
68 DO 73 I=2,KK3,2
73 B2N=B2N+ET(I)
GO TO 69
58 DO 91 I=1,KK3,2
91 B2N=B2N+ET(I)
DO 103 I=2,KK2,2
103 B2P=B2P+ET(I)
69 B2=B2P+B2N
70 CONTINUE
C

```



```

C COMPUTE B1 INDEX (ENERGY AREAS ENDING AT PX)
  KX1=KX-1
  B1=0.
  B1P=0.
  B1N=0.
  DO 105 I=1,KX1
  IF (ET(I).LT.0.) B1N=B1N+ET(I)
  IF (ET(I).GT.0.) B1P=B1P+ET(I)
105 CONTINUE
  IF (EX.LT.0.) B1N=B1N+EX
  IF (EX.GT.0.) B1P=B1P+EX
  B1=B1P+B1N
  RETURN
C GOES TO 110, IF RANN2 CANNOT BE COMPLETED DUE TO MANY SIGN CHANGES OF DT2
C OVER A SMALL PRESSURE INTERVAL, OR TOO MANY ENERGY AREAS (KK.GT.20).
110 EL=-999.
  B1=-999.
  B1P=-999.
  B1N=-999.
  B2=-999.
  B2P=-999.
  B2N=-999.
  WRITE (10,111)
111 FORMAT (1H," RANN2 SUBROUTINE DID NOT COMPLETE.")
  DP=DPSAVE ; RESTORE DP TO ORIGINAL VALUE, IF IT WAS CHANGED.
  RETURN
  END

```

*

*

PARAMETER LV=31 ; MAX NO. OF LEVELS + 1 EXTRA LEVEL FM BNDX
SUBROUTINE CNLMT(B2,EL,TEL,TCTOP,CTOP,EA,RLCL)

REV 01.00

MAR 1986 STONE, H. M. ERH FTS 649-5443
FORTRAN IV/ REV 5.57 DG ECLIPSE (S230) RDOS/ REV 6.17

PURPOSE

FINDS MAXIMUM PARCEL LEVEL (MPL) GIVEN EQUILIBRIUM LEVEL (MBS)
AND NET POSITIVE ENERGY BELOW EL. RAOB SOUNDING (P,TS,JNO)
IS AVAILABLE IN COMMON S.
SELECTS MPL TO THE NEAREST MILLIBAR, WHERE NEGATIVE ENERGY
ABOVE EL EQUALS OR EXCEEDS POSITIVE ENERGY BELOW EL.
PARCEL MUST BE SATURATED AT EL (RLCL.GE.EL) TO COMPUTE MPL.
(PARCEL COULD BE NON-SATURATED IN THE CASE OF DRY CONVECTION,
I.E., SUPERADIABATIC LAPSE RATE NEAR THE GROUND, THEN
NEGATIVE ENERGY AREA ABOVE. IN THIS CASE MPL IS NOT COMPUTED.)

ARGUMENT LIST

B2	- NET POSITIVE ENERGY BELOW EL (J/KG X 10)
EL	- EQUILIBRIUM LEVEL (MBS)
TEL	- TEMPERATURE AT EQUILIBRIUM LEVEL (DEG C)
TCTOP	- TEMPERATURE (DEG C) AT CTOP (MPL) LEVEL
CTOP	- MAXIMUM PARCEL LEVEL, MPL (MBS)
	CTOP=-999. ...UNDEFINED
	CTOP=-888. ...RAOB TOO LOW; P(JNO).EQ.100.
	CTOP=-777. ...RAOB TOO LOW; P(JNO).GT.100.
EA	- ENERGY BETWEEN EL AND CTOP (J/KG X 10)
RLCL	- LIFTING CONDENSATION LEVEL, LCL (MBS)

EXITS

STOP 'ERROR IN CNLMT, EL = P(JNO) ='
SHOULD NEVER OCCUR

```

C
C
COMMON/S/KDATE(3), I HOUR, JNO, JJNO, P(0:LV), TS(0:LV), TSD(0:LV)
THETA(T,P2,P1)=TX(P2/P1)**.2857142 ; DRY ADIABATIC (T,P1) TO (THETA,P2)
C
DP=10. ; 10MB PRESSURE STEP
IF(RLCL.LT.EL) GO TO 2 ; PARCEL IS NOT SATURATED
IF (B2.GT.0.) GO TO 1 ; COMPUTATION ONLY FOR POSITIVE B2
2
EA=-999.
TEL=-999.
TCTOP=-999.
CTOP=-999. ; CTOP NOT DEFINED
RETURN
1
C2=B2/28.704 ; DIVIDE OUT UNIT CONVERSION FACTOR FROM RANN2
DO 4 I=0,JNO
IF (EL-P(I)) 4,6,7
4
CONTINUE
TYPE "ERROR IN CNLMT, EL = ",EL," P(JNO) = ",P(JNO)
STOP ; SHOULD NEVER REACH THIS STATEMENT
6
TEL=TS(I)
GO TO 5
7
I=I-1
FACTOR=ALOG(EL/P(I+1))/ALOG(P(I)/P(I+1))
TEL=TS(I+1)+FACTOR*(TS(I)-TS(I+1))
5
CONTINUE
TH=THETA(TEL+273.16,1000.,EL)-273.16 ; POT TEMP DEG C
WTH=W0BF(TH)
WTC=W0BF(TEL)
THW=TH-WTH+WTC ; EQUIV WET BULB POT TEMP
P1=EL
DT1=0.
EA=0.
24
P2=P1-DP
II=0
IF (P(I+1)-P2) 9,9,8
8
IF ((I+1).LT.JNO) GO TO 16
IF(DP.EQ.1.) GO TO 15 ; RAOB NOT HIGH ENOUGH TO GET CTOP
DP=1.
GO TO 24
16
P2=P(I+1) ; RESET P2 TO LOWER SIG LVL
I=I+1
II=1
IF(P1.EQ.P2) GO TO 24 ; PREVIOUS STEP TERMINATED AT AN SGL LEVEL
9
PLOG1=ALOG(P(I)/P(I+1))
FACTORT=(TS(I)-TS(I+1))/PLOG1
TP=SATLFT(THW,P2)
PLOG2=ALOG(P2/P(I+1))
TE=TS(I+1)+PLOG2*FACTORT ; ENVIRONMENTAL TEMP AT P2
DP1=P1-P2
DT2=TE-TP
E=.5*(DT2+DT1)*ALOG(P1/P2)
EA=EA+E
C FOLLOWING 5 STATEMENTS FOR TEST
C GO TO 21
C EAC=EA*28.704
C WRITE (10,100) P1,P2,DP,B2,EAC
C100 FORMAT (1X,2F7.2,F4.0,2F8.3)
C21 CONTINUE
C
IF (EA-C2) 10,11,12

```

```

12 IF (DP.EQ.1.) GO TO 11
DP=1. ; REDUCE PRESSURE STEP TO 1MB
IF (II.EQ.1) I=I-1
EA=EA-E ; RESET ENERGY AREA ENDING AT P1
GO TO 24
10 P1=P2
DT1=DT2
GO TO 24
11 TCTOP=TP
CTOP=P2
EA=EA*28.784 ; CONVERSION FACTOR FOR J/KG X 10
RETURN
15 TCTOP=-777. ; -777 = RAOB TOO SHORT TO FIND CTOP LEVEL, P(JNO)>100
CTOP=-777.
IF (P(JNO).NE.100.) GO TO 14
TCTOP=-888.
14 CTOP=-888. ; -888 = RAOB TOO SHORT TO FIND CTOP LEVEL, P(JNO)=100
EA=EA*28.784 ; CONVERSION FACTOR FOR J/KG X 10
RETURN
END

```

*
SUBROUTINE HEIGHT (NS,PRES,HGT)

APR 1986 REV 01.10 ERH FTS 649-5443
 FORTRAN IV/ REV 5.57 STONE, H. M. RDOS/ REV 6.17
 DG ECLIPSE (S230)

PURPOSE

COMPUTES HEIGHT OF A PRESSURE SURFACE ACCORDING TO HEIGHTS
 OF STATION ELEVATION AND STANDARD LEVELS IN 'ZP' ARRAY.
 EXTRAPOLATION IS DONE ABOVE 100MBS AND BELOW SFC PRES P(0).

ARGUMENT LIST

NS

- NO. OF LEVELS TO BE USED IN ZP & SPP ARRAYS.
- SPP(0)=P(0) SFC PRESSURE, SPP(NS)= 100MBS.
- PRESSURE (MB) TO BE CONVERTED TO HEIGHT.
- HEIGHT (METERS) OF PRES

PRES
 HGT

COMMON/ZZ/ZP(0:10) ; ZP = HEIGHT OF STANDARD LEVELS
 COMMON/PS/SP(10),SPP(0:10) ; SPP= PRESSURE OF STANDARD LEVELS

DO 1 I=0,NS

IF (PRES-SPP(I)) 1,2,3

CONTINUE

I=I-1 ; 'I' IS INCREASED BY 1, WHEN LOOP FINISHED (PRES < 100MBS)

GO TO 4 ; EXTRAPOLATE UPWARD

HGT=ZP(I) ; PRES IS A STANDARD PRESSURE SFC

RETURN

IF (I.EQ.0) I=I+1 ; PRES > P(0) ... BLO SFC

HGT=ZP(I)+(ZP(I-1)-ZP(I))/ALOG(SPP(I-1)/SPP(I))*ALOG(PRES/SPP(I))

RETURN

END

*
FUNCTION JREAL (R)

SEP 1983
 FORTRAN IV/ REV 5.57
 PURPOSE

REV 01.00
 STONE, H. M.
 DG ECLIPSE (S230)

ERH FTS 649-5443
 RDOS/ REV 6.17

C
C
C
C

ROUNDS A REAL NUMBER TO AN INTEGER VALUE
ARGUMENT LIST
R - REAL NUMBER TO BE ROUNDED

RA=ABS(R)
JREAL=RA ; TRUNCATES POSITIVE R TO INTEGER
RD=RA-JREAL ; DECIMAL PORTION
IF (RD.GE..5) JREAL=JREAL+1
IF (R.LT.0.) JREAL=-JREAL ; CHANGE TO ORIGINAL SIGN
RETURN
END

*

*

OVERLAY OV1
PARAMETER LV=31
SUBROUTINE SHEAR(JNO1,P,TS,TSD,IW,ZW,DW,SU,SHR,IZ,NS)

C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C

REV 01.00
MAR 1986 STONE, H. M. ERH FTS 649-5443
FORTRAN IV/ REV 5.57 DG ECLIPSE (S230) RDOS/ REV 6.17
PURPOSE

COMPUTES DENSITY WEIGHTED MEAN WIND SFC TO 6KM ABOVE GROUND.
COMPUTES DENSITY WEIGHTED MEAN WIND SFC TO 500M AGL.
FROM THESE TWO ITEMS WIND SHEAR IS COMPUTED FOR USE IN
BULK RICHARDSON NO.

ARGUMENT LIST

JNO1 - NO. OF SIGNIFICANT LEVELS IN RAOB INCLUDING SURFACE. FROM TT88D DECODED SGL FILE.
P - ARRAY OF SGL PRESSURES IN RAOB
TS - ARRAY OF TEMPERATURES AT 'P' LEVELS
TSD - ARRAY OF DEWPOINTS AT 'P' LEVELS
IW - NO. OF WIND LEVELS IN DECODED RAOB, INCLUDING SURFACE LEVEL
ZW - HEIGHTS OF WIND LEVELS (METERS)
DW - WIND DIRECTIONS (DEGREES)
SU - WIND SPEEDS (KNOTS)
SHR - SHEAR OVER 6KMS (M/SEC)**2
SHR = -1, IF DECODED WINDS DO NOT GO TO 6KM
IZ - NO. OF MIDPOINTS TO 6KM BETWEEN DECODED LVLS
NS - NO. OF STANDARD LEVELS BETWEEN SFC AND 100 MBS

EXTERNALS

SDUV
PRES
RHOC

DIMENSION P(0:LV),TS(0:LV),TSD(0:LV),ZW(0:LV),DW(0:LV),SU(0:LV),
1 ZWI(0:LV),RHO(0:LV),Z2(0:0),ZR(0:0)
COMMON/PS/SP(10),SPP(0:10)
COMMON/ZZ/ZP(0:10)
ZTOP1=ZW(0)+6000. ; ELEVATION ABOVE SFC
ZTOP2=ZW(0)+500. ; ELEVATION ABOVE SFC
ZWI(0)=ZW(0)
IR=0
IZ=0 ; COUNTER FOR MIDPOINTS TO 6KM
IX=0 ; COUNTER FOR MIDPOINTS TO 500 METERS
KNOWND=1 ; INDICATES WINDS AVBL TO 6 KM
DO 1 I=1,IW
IZ=IZ+1
ZWI(I)=(ZW(I)+ZW(I-1))/2. ; MID POINT BTWN WIND LVLS

```

10 IF (ZTOP2-ZWI(1)) 10,10,11
   IF (IR.EQ.1) GO TO 9 ; COMPUTING MIDPOINTS ABOVE 500 METERS
   IR=1 ; INDICATOR THAT 500 METER LEVEL FOUND
   Z2(0)=ZTOP2
11 IX=IX+1
9 IF (ZTOP1-ZWI(1)) 3,2,1
1 CONTINUE
  KNOWND=0 ; WIND DOES NOT EXTEND TO 6KM
  GO TO 7
3 ZWI(1)=ZTOP1
2 CONTINUE
C CHANGE TO PRESSURE UNITS
  ZWI(0)=P(0) ; SFC PRES
  DO 4 I=1,IZ
4 ZWI(I)=PRES(ZWI(I),NS) ; CONVERT HEIGHT TO PRESSURE
  Z2(0)=PRES(Z2(0),NS)
  CALL RHOC(IZ,ZWI,P,TS,TSD,RHO) ; DENSITY OF ZWI LEVELS
C INTEGRATE TO 6KM
  UI=0.
  VI=0.
  DO 5 I=1,IZ
  CALL SDUV(SW(I-1),DW(I-1),U,V) ; GET U,V COMPONENTS
  DRHO=RHO(I-1)-RHO(I)
  UI=UI+U*DRHO
  VI=VI+V*DRHO
5 CONTINUE
  DRHO=RHO(0)-RHO(IZ)
  UI=UI/DRHO ; INTEGRAL TO 6KM
  VI=VI/DRHO ; INTEGRAL TO 6KM
C INTEGRATE TO 500 METERS
  CALL RHOC(0,Z2,P,TS,TSD,ZR) ; DENSITY AT 500 METER LEVEL
  RHO(IX)=ZR(0) ; DENSITY AT 500 METERS
  UIL=0.
  VIL=0.
  DO 6 I=1,IX
  CALL SDUV(SW(I-1),DW(I-1),U,V) ; GET U,V COMPONENTS
  DRHO=RHO(I-1)-RHO(I)
  UIL=UIL+U*DRHO
  VIL=VIL+V*DRHO
6 CONTINUE
  DRHO=RHO(0)-RHO(IX)
  UIL=UIL/DRHO ; INTEGRAL TO 500M
  VIL=VIL/DRHO ; INTEGRAL TO 500M
  U=(UI-UIL)*(UI-UIL)
  V=(VI-VIL)*(VI-VIL)
  SHR=.5*(U+V)*.2650098 ; SHEAR (1/2 U**2) - UNITS (M/SEC)**2
  GO TO 8
7 SHR=-1. ; INDICATOR THAT WINDS NOT AVAILABLE TO 6KM
8 RETURN
END

```

FUNCTION PRES(ZR,NS)

```

C REV 01.00
C MAR 1986 STONE, H. M. ERH FTS 649-5443
C FORTRAN IV/ REV 5.57 DG ECLIPSE (S230) RDOS/ REV 6.17
C PURPOSE
C COMPUTES A PRESSURE FOR A SPECIFIED HEIGHT (METERS), GIVEN A

```

C
C
C
C
C
C
C
C
C

PRESSURE/HEIGHT RELATIONSHIP FROM RAOB, OR STANDARD ATMOS.
INTERPOLATES BETWEEN STANDARD PRESSURE LEVELS.
EXTRAPOLATES BELOW SFC AND ABOVE 100 MBS.

ARGUMENT LIST

ZR - HEIGHT (METERS) TO BE CONVERTED TO PRES
NS - NO. OF STANDARD LEVELS IN ZP & SPP ARRAYS.
SPP(0)=P(0) SFC PRES, SPP(NT)=100MBS.

1
2
4
1
3

```
COMMON/PS/SP(10),SPP(0:10)
COMMON/ZZ/ZP(0:10)
DO 1 I=0,NS
IF (ZR-ZP(I)) 2,3,1
CONTINUE
I=I-1 ; I IS INCREASED BY 1, WHEN LOOP IS FINISHED (ZR ABV 100MBS)
GO TO 4
IF (I.GE.1) GO TO 4 ; ZR ABOVE SFC
I=I+1 ; EXTRAPOLATE DOWNWARD
PRES=ALOG(SPP(I-1)+(ZR-ZP(I-1))*(ALOG(SPP(I))-ALOG(SPP(I-1))))/
(ZP(I)-ZP(I-1))
PRES=EXP(PRES)
RETURN
PRES=SPP(I)
RETURN
END
```

*

*

PARAMETER LV=31
SUBROUTINE RHOC(IK,PI,P,TS,TSD,RHOI)

C
C
C
C
C
C
C
C
C
C
C

REV 01.00
MAR 1986 STONE, H. M. ERH FTS 649-5443
FORTRAN IV/ REV 5.57 DG ECLIPSE (S230) RDOS/ REV 6.17
PURPOSE

COMPUTES DENSITY FOR AN ARRAY OF PRESSURES, GIVEN A RAOB.

ARGUMENT LIST

IK - NO. OF PRESSURES IN PI ARRAY
PI - PRESSURE LEVELS FOR WHICH DENSITY TO BE COMPUTED
P - PRESSURES AT SGL LEVELS OF RAOB
TS - TEMPERATURES AT P LEVELS
TSD - DEWPOINTS AT P LEVELS
RHOI - DENSITIES (G/CM**3) AT PI LEVELS

17
13
14
15
16

```
DIMENSION PI(0:LV),P(0:LV),TS(0:LV),TSD(0:LV),RHOI(0:LV)
R=2.8704E+06 ; GAS CONSTANT, DRY AIR, ERG/G/K
J=0
DO 12 I=0,IK
IF (PI(I)-P(J)) 13,14,15
J=J+1
GO TO 17
TX=TS(J)+273.16 ; TEMPERATURE (K)
TDX=TSD(J) ; DEWPOINT (C)
GO TO 16
SLG=ALOG(P(J-1)/PI(I))/ALOG(P(J-1)/P(J))
TX=TS(J-1)-SLG*(TS(J-1)-TS(J))+273.16 ; TEMPERATURE (K)
TDX=TSD(J-1)-SLG*(TSD(J-1)-TSD(J)) ; DEWPOINT (C)
CONTINUE
W=WPROF(PI(I),TDX)*.001 ; MIXING RATIO (G/G)
TX=(1.+1.608*W)/(1.+W)*TX ; VIRTUAL TEMPERATURE (K)
```

12

```

RHO1(I)=PI(I)*1000./(R*TX) ; DENSITY (G/CM3)
CONTINUE
RETURN
END

```

*

```

SUBROUTINE SDUV (S,D,U,V)

```

REV 01.00

```

MAY 1982          STONE, H. M.          ERH  FTS 649-5443
FORTRAN IV/ REV 5.57  DG ECLIPSE (S230)  RDOS/ REV 6.17
PURPOSE

```

CONVERTS WIND SPEED AND DIRECTION TO U, V COMPONENTS

ARGUMENT LIST

```

S          - WIND SPEED (KNOTS)
D          - WIND DIRECTION (DEGREES)
U          - U COMPONENT (KNOTS)
V          - V COMPONENT (KNOTS)

```

PI=3.141592653

U=-S*SIN(D/100.*PI)

V=-S*COS(D/100.*PI)

RETURN

END

*

*

PARAMETER NRAOB=50

PARAMETER LV=31

```

SUBROUTINE GPT2(N,IRD,IST,RLCL,RLFC,A,B1,B2P0,B2N0,BRICH,SHR,P0,IT,IW)

```

REV 01.00

```

APR 1986          STONE, H. M.          ERH  FTS 649-5443
FORTRAN IV/ REV 5.57  DG ECLIPSE (S230)  RDOS/ REV 6.17
PURPOSE

```

WRITES VARIOUS STABILITY PARAMETERS ON LEFT SIDE OF HODOGRAPH AND CALLS SUBROUTINE GPT1 TO DRAW THE HODOGRAPH. WHEN COMPLETE IT STORES HODOGRAPH INTO AFOS PRODUCT "CCCHO", WHERE CCC IS 3 LETTER STATION ID.

ARGUMENT LIST

```

N          - TOTAL NUMBER OF RAOB STATIONS COMPUTED
IRD        - = 0, SIGNIFIES 'CCCSGLNNM' NOT AVAILABLE
IST        - ARRAY CONTAINING 3 LETTER ID FOR ALL RAOBS
RLCL       - LCL LEVEL (MB)
RLFC       - LFC LEVEL (MB)
A          - ARRAY CONTAINING ALL PARAMETERS IN 'EISTAB'
B1         - ENERGY INDEX EI (J/KG X 10)
B2P0       - + PART OF OLD EI2 WITH 0 ENTRAINMENT (J/KG)
B2N0       - - PART OF OLD EI2 WITH 0 ENTRAINMENT (J/KG)
BRICH      - BULK RICHARDSON NO.
SHR        - SHEAR OVER LOWEST 6KM OF ATMOS (M/SEC)*0.2
P0         - SURFACE PRESSURE (MB)
IT         - ARRAY CONTAINING DATE/TIME OF OBSERVATIONS
IW         - NO. OF WIND LEVELS TO FIRST LEVEL PAST 6KMS

```

EXTERNALS

TXX

GPT1

```

COMMON/CCL/PCCL,ETCCL,TS0,TS00,L,TSCCL,TCCL,TDCL,WAVG

```

```

COMMON/G/PP(8:28),ET(28),TJ(8:LV),DP,EFF,KMOD,KK
COMMON/TT/PT(8:LV),TST(8:LV),TSDT(8:LV)
COMMON/TITL1/KT(13),K03(5),K05(9),K06(5),K08(5),K09(5),K10(5),
1 K11(5),K12(5),K13(13),K14(6),K15(5),K16(5),K17(5),K18(5),K19(5),
2 K21(5),K22(5),K23(5),K24(5),K25(5),K26(5),K27(5),K28(5)
COMMON/TITLU/N1(2),N2(4),N3(6),N4(5),N5(5),N6(2),N7(2),N8(7),
1 N9(6),N10(4),N11(4)
COMMON/TITLM/KM(5)
COMMON/TXY/L2,LT
DIMENSION IST(NRAOB,2),ISC(7),KY(5),KKU(6),IT(16),A(12)
DIMENSION LX(5),LY(5)
DATA KT/'ENTRAINMENT = 60 PERCENT<00>'//
DATA KM/'MISSING <00>'//
DATA K03/'PMAX = <00>'//
DATA K05/'PARCEL FROM PMAX<00>'//
DATA K06/'EL = <00>'//
DATA K08/'LCL = <00>'//
DATA K09/'LFC = <00>'//
DATA K10/'EI = <00>'//
DATA K11/'EI+ = <00>'//
DATA K12/'EI- = <00>'//
DATA K13/'ENERGY CHANGE IN LAYERS <00>'//
DATA K14/'P1 P2 <00>'//
DATA K15/'LI = <00>'//
DATA K16/'KI = <00>'//
DATA K17/'SWI = <00>'//
DATA K18/'CCL = <00>'//
DATA K19/'C TMP = <00>'//
DATA K21/'WAVG = <00>'//
DATA K22/'B R N = <00>'//
DATA K23/'B+ = <00>'//
DATA K24/'B- = <00>'//
DATA K25/'SHR = <00>'//
DATA K26/'WMAX = <00>'//
DATA K27/'MPL = <00>'//
DATA K28/'P0 = <00>'//
DATA N1/'MB<00>'//
DATA N2/'HND FT<00>'//
DATA N3/'J/KG X 10 <00>'//
DATA N4/'(+ PART)<00>'//
DATA N5/'(- PART)<00>'//
DATA N6/'C <00>'//
DATA N7/'F <00>'//
DATA N8/'G/KG X 10-2 <00>'//
DATA N9/'(M/SEC)**2<00>'//
DATA N10/' M/SEC<00>'//
DATA N11/' XXX<00>'//
KY(1)=IST(N,1)
KY(2)=IST(N,2)
KY(3)=0 ; LOCATION IDENTIFIER
ISKIP=0 ; INDICATOR THAT B2P0 & B2N0 DEFINED
CALL TEXT(KY,1250,2950,3,1,0,0) ; MAIN TITLE
CALL TEXT(IT,1600,2950,3,1,0,0)
IF(IRD.EQ.0) GO TO 3 ; MISSING DATA
IF(SHR.EQ.-2.) GO TO 6 ; SHORT RAOB, NO INDICES COMPUTED
LT=2050
LP=75 ; SPACE BETWEEN LINES
L1=30
L2=L1+250
L3=L2+250

```



```

IF(B2P0.EQ.-9990.) ISKIP=1
CALL TEXT(K22,L1,LT,0,1,0,0) ; BRICH
IF(ISKIP.EQ.1.OR.SHR.LE.0.) GO TO 9 ; BRICH NOT COMPUTED
CALL TXX(BRICH)
GO TO 10
9 CALL TEXT(N11,L2,LT,1,1,0,0) ; PRINT 'XXX' MISSING
10 LT=LT-LP
CALL TEXT(K23,L1,LT,0,1,0,0) ; B+
CALL TXX(B2P0)
IF(ISKIP.EQ.1) CALL TEXT(N11,L2,LT,1,1,0,0) ; PRINT 'XXX' MISSING
CALL TEXT(N9,L3,LT,0,1,0,0)
LT=LT-LP
CALL TEXT(K24,L1,LT,0,1,0,0) ; B-
CALL TXX(B2N0)
IF(ISKIP.EQ.1) CALL TEXT(N11,L2,LT,1,1,0,0) ; PRINT 'XXX' MISSING
CALL TEXT(N9,L3,LT,0,1,0,0)
LT=LT-LP
CALL TEXT(K25,L1,LT,0,1,0,0) ; SHR
CALL TXX(SHR)
IF(SHR.EQ.-1.) CALL TEXT(N11,L2,LT,1,1,0,0) ; PRINT 'XXX' MISSING
CALL TEXT(N9,L3,LT,0,1,0,0)
LT=LT-LP
CALL TEXT(K26,L1,LT,0,1,0,0) ; WMAX
CALL TXX(A(2))
IF(ISKIP.EQ.1) CALL TEXT(N11,L2,LT,1,1,0,0) ; PRINT 'XXX' MISSING
CALL TEXT(N10,L3,LT,0,1,0,0)
LT=LT-LP
CALL TEXT(K06,L1,LT,0,1,0,0) ; EL. MB
CALL TXX(A(3))
CALL TEXT(N1,L3,LT,0,1,0,0)
LT=LT-LP
CALL TEXT(K06,L1,LT,0,1,0,0) ; EL. HND FT
CALL TXX(A(4))
CALL TEXT(N2,L3,LT,0,1,0,0)
LT=LT-LP
CALL TEXT(K27,L1,LT,0,1,0,0) ; MPL. HND FT
CALL TXX(A(5))
IF(A(5).EQ.-999.) CALL TEXT(N11,L2,LT,1,1,0,0) ; PRINT 'XXX' MISSING
CALL TEXT(N2,L3,LT,0,1,0,0)

```

C

```

LY(1)=LT-LP ; FOR BOX AROUND EI DATA

```

C

```

LT=LT-LP-LP
CALL TEXT(K05,L1,LT,0,1,0,0) ; TITLE
LT=LT-LP
CALL TEXT(KT,L1,LT,0,1,0,0) ; TITLE
LT=LT-LP
CALL TEXT(K28,L1,LT,0,1,0,0) ; P0
CALL TXX(P0)
CALL TEXT(N1,L3,LT,0,1,0,0)
LT=LT-LP
CALL TEXT(K03,L1,LT,0,1,0,0) ; PMAX
CALL TXX(PT(0))
CALL TEXT(N1,L3,LT,0,1,0,0)
LT=LT-LP
CALL TEXT(K08,L1,LT,0,1,0,0) ; LCL
CALL TXX(RLCL)
CALL TEXT(N1,L3,LT,0,1,0,0)
LT=LT-LP
CALL TEXT(K09,L1,LT,0,1,0,0) ; LFC

```

```

CALL TXX(RLFC)
IF(RLFC.EQ.0.) CALL TEXT(N11,L2,LT,1,1,0,0) ; PRINT 'XXX' MISSING
CALL TEXT(N1,L3,LT,0,1,0,0)
LT=LT-LP-LP
CALL TEXT(K10,L1,LT,0,1,0,0) ; EI
CALL TXX(B1)
CALL TEXT(N3,L3,LT,0,1,0,0)
LT=LT-LP
CALL TEXT(K11,L1,LT,0,1,0,0) ; EI+
CALL TXX(A(7))
CALL TEXT(N4,L3,LT,0,1,0,0)
LT=LT-LP
CALL TEXT(K12,L1,LT,0,1,0,0) ; EI-
CALL TXX(A(8))
CALL TEXT(N5,L3,LT,0,1,0,0)
LT=LT-LP-LP
CALL TEXT(K13,L1,LT,0,1,0,0) ; TITLE
LT=LT-LP
CALL TEXT(K14,L1,LT,0,1,0,0) ; TITLE
LT=LT-LP
CALL TEXT(N3,L3+200,LT,0,1,0,0)

```

C

```

KK1=KK-1 ; NO. OF LAYERS TO BE PRINTED OUT

```

```

JN=JREAL(PP(0))

```

```

CALL ISCR5(ISC,JN,+1)

```

```

DO 1 J=1,5

```

1

```

ISC(J)=ISC(J+2) ; SHIFT LEFT 2 SPACES

```

```

CALL TEXT(ISC,L1,LT,0,1,0,0)

```

```

L4=L3-50

```

```

DO 5 I=1,KK1

```

```

JP=JREAL(PP(I))

```

```

JT=JREAL(ET(I))

```

```

CALL ISCR5(ISC,JP,+1)

```

2

```

DO 2 J=1,5

```

```

ISC(J)=ISC(J+2) ; SHIFT LEFT 2 SPACES

```

```

CALL TEXT(ISC,L2,LT,0,1,0,0)

```

```

IF(I.LT.KK1) CALL TEXT(ISC,L1,LT-LP,0,1,0,0)

```

```

CALL ISCR5(ISC,JT,+1)

```

```

CALL TEXT(ISC,L4,LT,0,1,0,0)

```

```

LT=LT-LP

```

5

```

CONTINUE

```

```

LT=LT-LP*(9-KK1) ; THIS SECTION HAS 9 LINES, EVEN IF BLANK

```

C

```

LY(2)=LT+LP ; FOR BOX AROUND EI DATA

```

C

```

CALL TEXT(K15,L1,LT,0,1,0,0) ; LI

```

```

CALL TXX(A(10))

```

```

LT=LT-LP

```

```

CALL TEXT(K16,L1,LT,0,1,0,0) ; KI

```

```

CALL TXX(A(11))

```

```

IF(A(11).EQ.-999.) CALL TEXT(N11,L2,LT,1,1,0,0) ; PRINT 'XXX' MISSING

```

```

LT=LT-LP

```

```

CALL TEXT(K17,L1,LT,0,1,0,0) ; SWI

```

```

CALL TXX(A(12))

```

```

IF(A(12).EQ.-999.) CALL TEXT(N11,L2,LT,1,1,0,0) ; PRINT 'XXX' MISSING

```

```

LT=LT-LP-LP

```

```

CALL TEXT(K18,L1,LT,0,1,0,0) ; CCL

```

```

CALL TXX(PCCL)

```

```

IF(PCCL.EQ.-999.) CALL TEXT(N11,L2,LT,1,1,0,0) ; PRINT 'XXX' MISSING

```

```

CALL TEXT(N1,L3,LT,0,1,0,0)

```

```

LT=LT-LP
CALL TEXT(K19,L1,LT,0,1,0,0) ; C TEMP (C)
CALL TX(TSCCL)
IF(PCCL.EQ.-999.) CALL TEXT(N11,L2,LT,1,1,0,0) ; PRINT 'XXX' MISSING
CALL TEXT(N6,L3,LT,0,1,0,0)
LT=LT-LP
CALL TEXT(K19,L1,LT,0,1,0,0) ; C TEMP (F)
TSCCL=1.8*TSCCL+32. ; CONVERT TO DEG F
CALL TX(TSCCL)
IF(PCCL.EQ.-999.) CALL TEXT(N11,L2,LT,1,1,0,0) ; PRINT 'XXX' MISSING
CALL TEXT(N7,L3,LT,0,1,0,0)
LT=LT-LP
CALL TEXT(K21,L1,LT,0,1,0,0) ; WAVG
CALL TX(A(9))
CALL TEXT(N8,L3,LT,0,1,0,0)
C DRAW BOX AROUND EI DATA
LY(3)=LY(2)
LY(4)=LY(1)
LY(5)=LY(1)
LX(1)=0
LX(2)=0
LX(3)=1060
LX(4)=1060
LX(5)=0
CALL LINES(LX,LY,5,1,0)
6 CONTINUE
C HODOGRAPH COMPUTATION
IF(IW.GT.1) CALL GPT1(IW) ; MUST BE AT LEAST ONE LEVEL ABOVE SFC
4 CONTINUE
C
C CONSTRUCT KEY FOR HODOGRAPH
KY(3)="HO"
CALL UNPACK(KY,6,KKU)
DO 7 I=4,5
7 KKU(I)=KKU(I+1)
KKU(6)=0
CALL PACK(KKU,12,KY)
CALL Kfill(KY,IER)
IF(IER.NE.1) TYPE "Kfill IER = ",IER
CALL UTF(KY,"HMSGPH.01")
RETURN
3 CALL TEXT(KM,1600,1500,3,1,0,0) ; MISSING
GO TO 4
END

```

*

*

```

SUBROUTINE TX(TX)
C
C REV 1.00
C MAR 1986 STONE, H. M. ERH FTS 649-5443
C FORTRAN IV/ REV 5.57 DG ECLIPSE (S230) RDOS/ REV 6.17
C PURPOSE
C CONVERTS A REAL VARIABLE TO INTEGER AND WRITES TO A GRAPHIC
C ARGUMENT LIST
C TX - VARIABLE TO BE PLOTTED ON GRAPHIC
C
COMMON/TXY/L2,LT
DIMENSION ISC(7)
JN=JREAL(TX)

```

```
CALL ISCR5(ISC,JN,+1)
CALL TEXT(ISC,L2,LT,0,1,0,0)
RETURN
END
```

*

*
SUBROUTINE ISCR5 (ISC,JDAT,KSHIFT)

```
REV 01.00
MAR 1985 STONE, H. M. ERH FTS 649-5443
FORTRAN IV/ REV 5.20 DG ECLIPSE (S230) RDOS/REV 7.20
```

PURPOSE

CONVERTS 1 TO 5 DIGIT INTEGER (+ OR -) TO ASCII
CHARACTERS, SHIFTED TO LEFT OR RIGHT IN ARRAY 'ISC'
INTEGER IN ISC ENDS WITH ZERO (FOR USE WITH TEXT SUBROUTINE).

ARGUMENT LIST

```
ISC - OUTPUT ARRAY CONTAINING ASCII CHARACTERS
JDAT - INPUT, UP TO 5 DIGIT INTEGER
KSHIFT - -1 FOR SHIFT LEFT, +1 FOR SHIFT RIGHT
```

DIMENSION ISC(7)

```
ISC(1)=32 ; SPACE, SUPPRESS + SIGN
IF (JDAT.LT.0) ISC(1)=45 ; NEGATIVE SIGN
JDAT=IABS(JDAT) ; USE ABSOLUTE VALUE OF JDAT
IS=JDAT
```

IK=0

JK=0

IDIV=10000

DO 1 I=2,5

ISC(I)=IS/IDIV ; SEPERATING INDIVIDUAL DIGITS

IS=IS-ISC(I)*IDIV ; REMAINING NUMBER

IF (ISC(I).NE.0) JK=1 ; JK=1 1ST TIME NON-ZERO ENCOUNTERED

IF (JK.EQ.1) GO TO 6

IF (ISC(I).EQ.0) IK=IK+1 ; COUNTING NUMBER OF LEADING ZERO DIGITS

ISC(I)=ISC(I)+48 ; CONVERT TO ASCII

IDIV=IDIV/10

CONTINUE

ISC(6)=IS+48 ; UNITS DIGIT, CONVERT TO ASCII

ISC(7)=0 ; SET TO ZERO FOR TEXT SUBROUTINE

IF (IK.EQ.0) RETURN ; 5 DIGIT NUMBER, NO SHIFT NECESSARY

IF (KSHIFT.EQ.1) GO TO 2 ; SHIFT RIGHT

SHIFT LEFT

IK7=7-IK

DO 4 I=2,IK7

ISC(I)=ISC(I+IK)

IK8=IK7+1

DO 5 I=IK8,7

ISC(I)=32 ; SET END OF ARRAY TO SPACES

RETURN

SHIFT RIGHT

ISC(IK+1)=ISC(1) ; SHIFT SIGN TO RIGHT

DO 3 I=1,IK

ISC(I)=32 ; SET LEADING DIGITS TO SPACE

RETURN

END

*

*

PARAMETER LV=31 ; NO. OF WIND LEVELS
SUBROUTINE GPT1(IW)

```
C          REV 01.00
C          MAR 1966 - STONE, H. M. ERH FTS 649-5443
C          FORTRAN IV/ REV 5.57 DG ECLIPSE (S230) RDOS/ REV 6.17
C          PURPOSE
C          PLOTS HODOGRAPH TO 6KM OR FIRST REPORTED LEVEL PAST 6KM
C          ARGUMENT LIST
C          IW          - NO. OF WIND LEVELS TO BE PLOTTED
C          EXTERNALS
C          DIAM
C          ISCR5
C          JREAL
C
C          COMMON/CCL/PCCL,ETCCL,TS0,TS00,L,TSCCL,TCCL,TCCL,WAVG
C          COMMON/WND/ZW(0:LV),DW(0:LV),SW(0:LV)
C          UNITS: ZW:METERS DW:DEGREES SW:KNOTS
C          COMMON/TITLH/KVS(3),KVN(3),KUE(3),KUW(3),NL(5),NU(19)
C          DIMENSION LX(2),LY(2),ISC(7),IV(2),KR(2),KL(2)
C          DIMENSION V(2,0:LV),VMIN(2),VMAX(2),JVL(2,20)
C          DATA KVS/'100 <00>' /
C          DATA KVN/'360 <00>' /
C          DATA KUE/'090 <00>' /
C          DATA KUW/'270 <00>' /
C          DATA NL/'L = <00>' /
C          DATA NU/'UNITS : KNOTS, LVLS : THSD FT (MSL) <00>' /
C          TINV=14. ; MAXIMUM NO. OF INTERVALS ON U, V AXES
C          KM=100 ; MARGIN OFFSET
C          KL(1)=1195+KM ; LEFT BORDER X
C          KR(1)=4095-KM-120 ; RIGHT BORDER X
C          KL(2)=0+KM+60 ; LEFT BORDER Y
C          KR(2)=2900-KM-60 ; RIGHT BORDER Y
C          ZW(0)=0. ; SFC LEVEL
C          DO 1 I=1,IW
C          ZW(I)=ZW(I)/304.8 ; METERS TO THSDS OF FEET
1          CONTINUE
C          DETERMINE COORDINATE SYSTEM
C          DO 3 J=1,2
C          VMIN(J)=-1.E-50 ; ZERO MUST ALWAYS APPEAR ON HODOGRAPH, SO
3          VMAX(J)=+1.E-50 ; THESE VALUES DO HODOGRAPH AXES CORRECTLY
C          DO 4 I=0,IW
C          CALL SDUV(SW(I),DW(I),V(1,I),V(2,I)) ; U,V COMPONENTS (KTS)
C          WRITE (10,100) I,ZW(I),DW(I),SW(I),V(1,I),V(2,I) ; TEST ONLY!!!
C100          FORMAT (1X,12,3F10.0,2F12.2)
C          DO 4 J=1,2
C          IF (V(J,I).LT.VMIN(J)) VMIN(J)=V(J,I) ; FIND MIN
4          IF (V(J,I).GT.VMAX(J)) VMAX(J)=V(J,I) ; FIND MAX
C          WRITE (10,101) IW,(VMAX(J),J=1,2),(VMIN(J),J=1,2) ; TEST ONLY!!!!
C101          FORMAT (1X,12,4F12.2)
C          FIND RANGE OF U & V
C          DV1=VMAX(1)-VMIN(1)
C          DV2=VMAX(2)-VMIN(2)
C          IF(DV1-DV2) 8,9,9
8          IB=2
C          VB=DV2
C          GO TO 10
9          IB=1
C          VB=DV1
10         IN=2
C          T=VB/IN
```

```

IF(T.LT.TINV) GO TO 5 ; CORRECT INTERVAL IS 2
IN=5
T=VB/IN
IF(T.LT.TINV) GO TO 5 ; CORRECT INTERVAL IS 5
IN=10 ; USE INTERVAL 10
5 CONTINUE
DO 6 J=1,2
TML=VMIN(J)/IN.
ML=TML ; TRUNCATE TML
FT=TML-ML
IF(FT.NE.0..AND.ML.LE.0) ML=ML-1
IV(J)=1
JVL(J,1)=ML*IN
7 IV(J)=IV(J)+1
JVL(J,IV(J))=JVL(J,IV(J)-1)+IN
IF(JVL(J,IV(J)).LT.VMAX(J)) GO TO 7
6 CONTINUE
RKR=KR(IB)
RKL=KL(IB)
FACTOR=(RKR-RKL)/(JVL(IB,IV(IB))-JVL(IB,1))
IF(IB.EQ.1) GO TO 11 ; CHANGE 'IB' TO THE SMALLER RANGE.
IB=1
GO TO 12
11 IB=2
12 CONTINUE
C SHIFT HODOGRAPH SO THAT SMALLER RANGE IS IN CENTER OF PLOTTING SPACE
MARSHIFT=.5*FACTOR*(JVL(IB,IV(IB))-JVL(IB,1))
KCENTER=.5*(KL(IB)+KR(IB)) ; CENTER OF PLOTTING SPACE
KL(IB)=KCENTER-MARSHIFT
KR(IB)=KCENTER+MARSHIFT
C
C DRAW COORDINATE AXES
C Y AXIS
LX(1)=KL(1)+FACTOR*(0.-JVL(1,1)) ; X COORD
LX(2)=LX(1)
LY(1)=KL(2)
LY(2)=KR(2)
CALL LINES(LX,LY,2,1,0)
C PUT ON REFERENCE MARKERS
LX(2)=LX(1)+20
LX1=LX(1)-235 ; OFFSET FOR TEXT
IJ=IV(2)
DO 14 I=1,IJ
LY(1)=KL(2)+FACTOR*(JVL(2,I)-JVL(2,1))
LY(2)=LY(1)
CALL LINES(LX,LY,2,1,0)
IF(I.EQ.1.OR.I.EQ.IJ) GO TO 14
IF(JVL(2,I).EQ.0) GO TO 14 ; DON'T PRINT '0' ON Y AXIS
CALL ISCR5(ISC,JVL(2,I),+1)
CALL TEXT(ISC,LX1,LY(1)-15,0,1,0,0)
14 CONTINUE
CALL TEXT(KVS,LX(1)-110,KR(2)+20,3,1,0,0) ; 180 DEG
CALL TEXT(KVN,LX(1)-110,KL(2)-80,3,1,0,0) ; 360 DEG
C
C X AXIS
LX(1)=KL(1)
LX(2)=KR(1)
LY(1)=KL(2)+FACTOR*(0.-JVL(2,1)) ; Y COORD
LY(2)=LY(1)
CALL LINES(LX,LY,2,1,0)

```

```

C PUT ON REFERENCE MARKERS
  LY(1)=LY(1)+20
  LY1=LY(1)-80 ; OFFSET FOR TEXT
  IJ=IV(1)- ; NO. OF INTERVAL MARKERS
  DO 15 I=1,IJ
  LX(1)=KL(1)+FACTOR*(JVL(1,I)-JVL(1,1))
  LX(2)=LX(1)
  CALL LINES(LX,LY,2,1,0)
  IF(I.EQ.1.OR.I.EQ.IJ) GO TO 15
  IF(JVL(1,I).EQ.0) GO TO 15 ; DON'T PRINT '0' ON X AXIS
  CALL ISCR5(ISC,JVL(1,I),+1)
  CALL TEXT(ISC,LX(1)-185,LY1,0,1,0,0)
15 CONTINUE
  CALL TEXT(KUW,LX(1)+10,LY(1)-60,3,1,0,0) ; 270 DEG
  CALL TEXT(KUE,KL(1)-220,LY(1)-60,3,1,0,0) ; 090 DEG

```

```

C
C DRAW HODOGRAPH
C

```

```

  LX(2)=KL(1)+FACTOR*(0,-JVL(1,1)) ; ORIGIN
  LY(2)=KL(2)+FACTOR*(0,-JVL(2,1))
  ISIZE=15 ; SIZE OF DIAMOND BOX AROUND DATA POINT
  DO 16 I=0,IW
  LX(1)=LX(2)
  LY(1)=LY(2)
  LX(2)=KL(1)+FACTOR*(V(1,I)-JVL(1,1))
  LY(2)=KL(2)+FACTOR*(V(2,I)-JVL(2,1))
  CALL LINES(LX,LY,2,1,0)
  CALL DIAM(LX(2),LY(2),ISIZE) ; DRAW DIAMOND AROUND DATA POINT
  JZW=JREAL(ZW(1))
  CALL ISCR5(ISC,JZW,-1)
  CALL TEXT(ISC,LX(2)-20,LY(2)+30,0,1,0,0)
16 CONTINUE
C PRINT NO. OF WIND LEVELS IN LOWER RIGHT CORNER
  IW=IW+1 ; NO. OF LEVELS INCLUDING SFC
  CALL ISCR5(ISC,IW,+1)
  NL(3)=ISC(5)
  NL(4)=ISC(6)
  CALL TEXT(NL,3840,0,0,1,0,0)
C PRINT UNITS IN UPPER RIGHT CORNER
  CALL TEXT(NU,2630,2850,0,1,0,0)
  RETURN
  END

```

*

*

SUBROUTINE DIAM (IX,IY,ISIZE)

REV 01.00

MAR 1986

STONE, H. M.

ERH FTS 649-5443

FORTRAN IV/ REV 5.57

DG ECLIPSE (S230)

RDOS/ REV 6.17

PURPOSE

DRAW DIAMOND BOX AROUND A DATA POINT ON A GRAPHIC

ARGUMENT LIST

IX

- X COORDINATE OF DATA POINT

IY

- Y COORDINATE OF DATA POINT

ISIZE

- DETERMINES SIZE OF BOX, 15 = ABOUT 2MM

DIMENSION LX(5),LY(5)

LX(1)=IX-ISIZE

LY(1)=IY

```

LX(2)=IX
LY(2)=IY-ISIZE
LX(3)=IX+ISIZE
LY(3)=IY
LX(4)=IX
LY(4)=IY+ISIZE
LX(5)=LX(1)
LY(5)=LY(1)
CALL LINES(LX,LY,5,1,0)
RETURN
END

```

*

*

```

OVERLAY OV2
PARAMETER NRAOB=50 ; MAX NO. OF STATIONS ON GRAPHIC
PARAMETER NT=2 ; MAX NO. OF PLOT GRAPHICS
SUBROUTINE FINH (N, IXX, IYY, JR, JL, IST, IT, MV)

```

REV 01.00

```

MAR 1986          STONE, H. M.          ERH   FTS 649-5443
FORTRAN IV/ REV 5.57  DG ECLIPSE (S230)  RDOS/ REV 6.17
PURPOSE

```

```

LAST SUBROUTINE CALL BY CONVECT.
INSERTS HEADING AND ENDING ON FILE 'INDEXX' AND STORES IT
INTO AFOS PRODUCT 'EISTAB'. CREATES 'EIS' & 'EIT' GRAPHICS.

```

ARGUMENT LIST

N	- NO. RAOB STATIONS THAT HAVE BEEN COMPUTED
IXX	- ARRAY OF 'X' MAP COORDINATES FOR RAOB STNS
IYY	- ARRAY OF 'Y' MAP COORDINATES FOR RAOB STNS
JR	- ARRAY OF VARIABLES FOR PLOTTING ON RIGHT HAND SIDE OF STN MODEL, GRAPHIC EIS & EIT
JL	- ARRAY OF VARIABLES FOR PLOTTING ON LEFT HAND SIDE OF STN MODEL, GRAPHIC EIS & EIT
IST	- ARRAY OF 3 LETTER IDS OF RAOB STATIONS
IT	- DATE/TIME ARRAY FOR GRAPHIC TITLE
MV	- ARRAY OF CODE NOS. FOR VARIARBLS ON EIS & EI

EXTERNALS

GPT

CHANNELS/FILES

ICHN - INDEXX FOR 'EISTAB' AFOS PRODUCT

```

COMMON/H/IHDR1(11),KEY(5)
DIMENSION IXX(NRAOB),IYY(NRAOB),JR(NRAOB,NT),JL(NRAOB,NT),
1  IST(NRAOB,2)
DIMENSION IT(16),MV(4)
DATA IHDR1/'  ISTAB000',177777K,177777K,'70',142600K,6412K/
DATA KEY/'EISTAB'/
IEND=101603K ; ENDING FOR AFOS PRODUCT
CALL KFULL (KEY,IER)
IF(IER.NE.1) TYPE "KFULL, IER = ",IER
IHDR1(1)=KEY(1)
IHDR1(2)=KEY(2)

```

INSERT HEADING AND ENDING ON INDEXX

```

CALL GCHN (ICHN,IER) ; GET RDOS CHANNEL
CALL OPENN (ICHN,"INDEXX",0,IER)
CALL WRS (ICHN,IHDR1,22,IER) ; HEADER INSERTION
CALL KLOSE (ICHN,IER)
CALL GCHN (ICHN,IER) ; GET RDOS CHANNEL

```



```

CALL OPENA (ICHN,"INDEXX",0,IER) ; OPEN FOR APPENDING
CALL WRS (ICHN,IEND,2,IER) ; ENDING FOR AFOS PRODUCT
CALL KLOSE (ICHN,IER)
CALL FSTORE ("INDEXX",0,IER) ; STORE INTO EISTAB

```

```

C
C CREATE GRAPHIC EIS
  K=1
  CALL GPT(N,IXX,IYY,JR,JL,IST,IT,K,MV)
C CREATE GRAPHIC EIT
  K=2
  CALL GPT(N,IXX,IYY,JR,JL,IST,IT,K,MV)
  CALL FORKP ("CONVECT","EISTAB & EIS",IER) ; TURN ON ALERT LIGHT
  DO 61 I=1,1000
  CALL DFILW ("INDEXX",IER) ; DELETE INDEXX FILE
  IF (IER.EQ.1) GO TO 32
61 CONTINUE
  IF (IER.NE.1) TYPE "INDEXX FILE NOT DELETED, IER = ",IER
32 DO 37 I=1,1000
  CALL DFILW ("HMSGPH.01",IER) ; DELETE GRAPHIC FILE
  IF (IER.EQ.1) GO TO 91 ; THIS LOOP IS NECESSARY, FOR SLOW CLOSING
37 CONTINUE
  IF (IER.NE.1) WRITE (10,36) IER
36 FORMAT (1H,"IER = ",14," HMSGPH.01 NOT DELETED - FINI, STATEMENT 36")
91 RETURN
  END

```

*

```

*
PARAMETER NRAOB=50 ; MAX. NO. OF STATIONS IN PLOT
PARAMETER NT=2 ; MAX NO. OF PLOT GRAPHICS
SUBROUTINE GPT(N,IXX,IYY,JR,JL,IST,IT,K,MV)

```

```

C REV 01.00
C MAR 1986 STONE, H. M. ERH FTS 649-5443
C FORTRAN IV/ REV 5.57 DG ECLIPSE (S230) RDOS/ REV 6.17
C PURPOSE

```

```

C PLOTS GRAPHICS EIS AND EIT
C ARGUMENT LIST

```

```

C N - NO. OF RAOB STATIONS FOR PLOTTING
C IXX - ARRAY OF 'X' MAP COORDINATES OF RAOB STNS
C IYY - ARRAY OF 'Y' MAP COORDINATES OF RAOB STNS
C JR - ARRAY OF VARIABLES FOR PLOT ON RIGHT HAND
C SIDE OF STATION MODEL
C JL - ARRAY OF VARIABLES FOR PLOT ON LEFT HAND
C SIDE OF STATION MODEL
C IST - ARRAY OF 3 LETTER IDS FOR RAOB STATIONS
C IT - DATE/TIME ARRAY FOR TITLE
C K - INDICATOR, K = 1 ... EIS, K = 2 ... EIT
C MV - ARRAY SPECIFYING VRBL FOR PLOTTING...SAME
C AS COLUMN NUMBERS ON 'EISTAB' PRODUCT.

```

```

C EXTERNALS
C ISCRS
C

```

```

COMMON/H/IHDR1(11),KEY(5)
DIMENSION IXX(NRAOB),IYY(NRAOB),JR(NRAOB,NT),JL(NRAOB,NT),
1 IST(NRAOB,2)
DIMENSION ISC(7),JS(3),IT(16),MV(4),JT(8)
COMMON/TITLE/JTT(3,12),INON(3)
DATA JTT/'POMX WMAX EL(MB)EL(FT)MPL --TROP EI+ EI- WAVG ',
1 "LI KI SWI "/

```

```

DATA INON/"NONE<00>"/
IF(K.EQ.2.AND.MV(4).EQ.0) GO TO 12 ; NO VARIABLE FOR EIT
L=2*K-1 ; INDICATOR FOR WHICH VARIABLE TO PLOT
LL=L+1
DO 10 I=1,N
IX=IX(I)
IY=IY(I)
IF(JR(I,K).EQ.-999.AND.JL(I,K).EQ.-999) GO TO 1 ; LEFT & RIGHT MISG
IF(JR(I,K).EQ.-999.AND.MV(L).EQ.0) GO TO 1 ; SINGLE VRBL PLOT MISG
JDAT=JR(I,K)
CALL ISCR5(ISC,JDAT,-1) ; CONVERT JR TO ASCII
IYOF=12 ; Y OFFSET FOR PLOTTING
CALL TEXT (ISC,IX,IY,1,2,5,IYOF) ; PLOT JR
IF(MV(L).EQ.0) GO TO 3 ; NO LEFT VARIABLE TO PLOT
JDAT=JL(I,K)
CALL ISCR5(ISC,JDAT,+1) ; CONVERT JL TO ASCII
DO 5 J=1,5
5 ISC(J)=ISC(J+2) ; SHIFT 2 DIGITS TO LEFT
CALL TEXT (ISC,IX,IY,1,2,-30,IYOF) ; PLOT JL
3 ISC(2)=14 ; STATION SYMBOL, CLEAR
GO TO 2
1 ISC(2)=5 ; MISSING DATA
2 ISC(1)=22K ; START SPECIAL SYMBOLS
ISC(3)=21K ; END SPECIAL SYMBOLS
DO 4 J=4,7
4 ISC(J)=0
IF (JR(I,K).GT.0.AND.JR(I,K).NE.-999) ISC(2)=3 ; + PLOT OVERCAST
CALL TEXT (ISC,IX,IY,1,1,0,0) ; PLOT STATION SYMBOL
JS(1)=IST(I,1)
JS(2)=IST(I,2)
JS(3)=0
CALL TEXT (JS,IX,IY,1,1,-7,-10) ; PLOT STATION ID
10 CONTINUE
C CONSTRUCT 1ST LINE OF TITLE
IF(MV(L).EQ.0) GO TO 7
DO 6 I=1,3
6 JT(I)=JTT(I,MV(L)) ; VARIABLE PLOTTED LEFT
JT(4)="/"
DO 8 I=1,3
8 JT(I+4)=JTT(I,MV(LL)) ; VARIABLE PLOTTED RIGHT
JT(8)=0 ; TERMINATOR
C ADJUST TITLE FOR 'EIS' GRAPHIC
IF(K.EQ.1) JT(6)="/" ; CHANGE 'EI+' TO 'EI'
GO TO 11
7 DO 9 I=1,3 ; ONE VARIABLE ON RIGHT
9 JT(I)=JTT(I,MV(LL))
JT(4)=0 ; TERMINATOR
C PRINT TITLE IN LOWER RIGHT CORNER OF GRAPHIC
11 CALL TEXT (JT,2600,550,3,1,0,0) ; FIRST LINE OF TITLE
GO TO 13
12 CALL TEXT(INON,1600,1500,3,1,0,0) ; WRITE 'NONE'
13 CALL TEXT (IT,2600,450,3,1,0,0) ; DATE/TIME LINE OF TITLE
IF(K.EQ.1) CALL UTF("NMGPH.EIS","HMSGPH.01")
IF(K.EQ.2) CALL UTF("NMGPH.EIT","HMSGPH.01")
RETURN
END

```

*

*

```

FUNCTION DPTOF(EW)
COMPUTE DEWPOINT, DPT, IN DEGREES C GIVEN WATER VAPOR PRESSURE (MB)
CREATE TOLERANCE TO DEGREE DESIRED
TOL=0.00010
IF (EW-0.21382876E-09) 20,20,30
20 DPTOF=10000.
RETURN
30 IF (1013.0-EW) 20,100,100
CREATE GUESS BY INVERTING TETEN-S FORMULA
100 X=ALOG(EW/6.1078)
BOT=17.269388-X
DPTOF=(237.3*X)/BOT
BOT=BOT*EW
DELT1=0.
200 EDP=VAPFW(DPTOF)
CORRECT GUESS BY DERIVATIVE OF TEMPERATURE WITH RESPECT TO VAPOR PRES.
CALCULATED FROM INVERSE OF TETEN-S FORMULA
DTDE=(DPTOF+237.3)/BOT
DELT=DTDE*(EW-EDP)
DPTOF=DPTOF+DELT
CHECK THAT ITERATION IS NOT IN AN ENDLESS CYCLE, A RARE SITUATION
C IF NEEDED, CHANGE -TOL- AND EXIT
DM=DELT-DELT1
IF (ABS(DM).GE.1.E-7) GO TO 10 ; IF DM VERY SMALL, ITERATION IS ENDLESS
TOL=ABS(DELT)
10 DELT1=-DELT
CHECK TO SEE IF ANSWER CLOSE ENOUGH, IF NOT ITERATE OVER CORRECTION
IF (ABS(DELT)-TOL) 300,300,200
CHANGE SO DEWPOINT IS ALWAYS LESS THAN THE TEMP.
COMPATIBILITY WITH TOL IS FORCED
300 DPTOF=DPTOF-TOL
RETURN
END

```

*

*

```

FUNCTION SATLFT (THM,P)
COMPUTES TEMPERATURE (DEG C) WHERE THETA MOIST (DEG C) CROSSES P (MB)
CONSIDER THE EXPONENTIAL FOR POTENTIAL TEMPERATURE AS ROCP
ROCP=0.28571428
IF (ABS(P-1000.0)-0.0010) 100,100,200
100 SATLFT=THM
RETURN
200 PURP=(P/1000.0)**ROCP
COMPUTE TEMPERATURE OF DRY ADIABATIC LIFT FOR FIRST GUESS
TONE=(THM+273.16)*PURP-273.16
CONSIDER PSEUDO-ADIABAT, EW1, THROUGH TONE AT P.
COMPUTE EONE=EW1-THM
EONE=WOBF(TONE)-WOBF(THM)
RATE=1.0
GO TO 330
300 CONTINUE
CONTRIBUTION TO ITERATION IS CHANGE IN T
CORRESPONDING TO CHANGE IN E
RATE=(TTWO-TONE)/(ETWO-EONE)
TONE=TTWO
EONE=ETWO
330 CONTINUE
COMPUTE ESTIMATED SATLIFT, TTWO

```

```

TTWO=TONE-EONE*RATE
CONSIDER PSEUDO-ADIABAT, EW2, THROUGH TTWO AT P.
COMPUTE ETWO=EW2-THM
ETWO=(TTWO+273.16)/PWRP-273.16
ETWO=ETWO+W0BF(TTWO)-W0BF(ETWO)-THM
CORRECTION TO TTWO IS EOR
EOR=ETWO*RATE
IF(ABS(EOR)-0.1000) 400,400,300
400 SATLFT=TTWO-EOR
RETURN
END

```

*

```

FUNCTION TCONOF(TEMP,DEWPT)
COMPUTES CONDENSATION TEMPERATURE (DEGREES CENT) BY LIFTING
S=TEMP-DEWPT
CONSIDER TEMP AND DEWPT TO BE LIKE UNITS (C OR K)
T=TEMP
IF(100.-TEMP) 4,5,5
4 T=TEMP-273.16
COMPUTE CURVE FIT IN MOST EFFICIENT MANNER
5 DLT=S*(1.2185+0.001278*T+S*(-0.002190+11.73E-6*S-5.20E-6*T))
TCONOF=T-DLT
RETURN
END

```

*

```

FUNCTION VAPFW(T)
COMPUTE SATURATION VAPOR PRESSURE OVER WATER, VAPFW, IN MBS.
CONSIDER T(TEMPERATURE) IN DEGREES C OR DEGREES K.
X=T
IF (100.0-X) 3,4,4
3 X=X-273.16
CURVE FIT FOR RANGE -50 < T < 100 DEGREES C.
4 POL = 0.99999683 E-00 + X *(-0.90826951 E-02 +
1 X *(0.78736169 E-04 + X *(-0.61117958 E-06 +
2 X *(0.43884187 E-08 + X *(-0.29883885 E-10 +
3 X *(0.21874425 E-12 + X *(-0.17892321 E-14 +
4 X *(0.11112018 E-16 + X *(-0.30994571 E-19)))))))))
POL=POL*POL
POL=POL*POL
VAPFW=6.107800/(POL*POL)
RETURN
END

```

*

```

FUNCTION WPROF(P,TD)
COMPUTE MIXING RATIO (G/KG)...DEWPOINT (DEGREES C OR K)...PRESSURE (MB)
T=TD
IF (100.-T) 3,4,4
3 T=T-273.16
CURVE FIT CORRECTION FOR NON-IDEAL GAS
4 X=0.0200*(T-12.5+7500.0/P)
WFW=1.+0.0000045*P+0.00140***X

```

```

COMPUTE ACCORDING TO STANDARD FORMULA
FUESW=WFLW*VAPFJ(T)
WPROF=621.97*(FUESW/(P-FUESW))
RETURN
END

```

*

*

```

FUNCTION WOFB(T)
COMPUTE BY DOUBLE ASYMPTOTIC APPROXIMATION
CONSIDER SEPARATELY IF .GT. OR .LE. 20 DEG.
CENT. FOR ALL TEMPS...THETW=THETA-WOFB(THETA)+WOFB(TEMPCON)
CENT. FOR ALL TEMPS...THETM=THETA-WOFB(THETA)+WOFB(TEMP)

```

```

X=T-20.0

```

```

IF(X) 10,10,20

```

```

10 CONTINUE

```

```

CURVE FIT FOR COOL TEMPERATURE RANGE

```

```

POL=1.000+X*(-8.8416605E-3+X*(1.4714143E-4+X*(-9.6719890E-7

```

```

1 +X*(-3.2687217E-8+X*(-3.8598073E-10))))

```

```

POL=POL*POL

```

```

WOFB=15.130/(POL*POL)

```

```

RETURN

```

```

20 CONTINUE

```

```

CURVE FIT FOR WARMER TEMPERATURES

```

```

POL=1.000+X*(3.6182989E-3+X*(-1.3683273E-5+X*(4.9618922E-7

```

```

1 +X*(-6.1859365E-9+X*(3.9481551E-11+X*(-1.2588129E-13

```

```

2 +X*(1.6688280E-16))))))

```

```

POL=POL*POL

```

```

WOFB=29.930/(POL*POL)+0.9600*X-14.800

```

```

RETURN

```

```

END

```

INDEX OF PROGRAM LISTINGS

<u>PROGRAM</u>	<u>PAGE</u>
BNDX.....	41
BNSCH.....	35
CCL1.....	37
CNLMT.....	49
CONVECT.....	23
DIAM.....	63
DPTOF.....	67
FINH.....	64
GPT.....	65
GPT1.....	61
GPT2.....	55
HEIGHT.....	50
INDX1.....	39
ISCR5.....	60
JREAL.....	50
MTITL.....	32
PRES.....	53
PRLIM.....	30
RANN2.....	42
RHOC.....	54
RSTND.....	34
SATLFT.....	67
SDUV.....	55
SHEAR.....	52
TCONOF.....	68
TXX.....	59
VAPFW.....	68
WMROF.....	68
WOBF.....	69

Eastern Region Computer Programs and Problems (Continued)

- 19 **Verification of Asynchronous Transmissions.** Lawrence Cedrone, March 1984. (PB84 189885)
- 20 **AFOS Hurricane Plotter.** Charles Little, May 1984. (PB84 199629)
- 21 **WARN - A Warning Formatter.** Gerald G. Rigdon, June 1984. (PB84 204551)
- 22 **Plotting TDL Coastal Wind Forecasts,** Paula Severe, June 1984 (Revised) (PB84-220789)
- 23 **Severe Weather Statistics STADTS Decoder (SWX) and Plotter (SWY),** Hugh M. Stone, June 1984. (PB84-213693)
- 24 **WXR,** Harold Opitz, August 1984. (PB84-23722) (Revised August 1985, PB86 100815/AS)
- 25 **FTASUM: Aviation Forecast Summaries,** Matthew Peroutka, August 1984. (PB85-112977)
- 26 **SAOSUM: A Short Summary of Observations.** Matthew Peroutka, October 1984. (PB85-120384)
- 27 **TRAJ - Single Station Trajectory Plot,** Tom Niziol, December 1984. (PB85-135002)
- 28 **VIDTEX,** Gerald G. Rigdon, February 1985 (PB85-175669/AS)
- 29 **ISENTROPIC PLOTTER,** Charles D. Little, February 1985 (PB85-175651/AS)
- 30 **CERR: An Aviation Verification Program,** M. Peroutka, April 1985. (PB85-204824/AS)
- 31 **Correlation and Regression Equation Program - REGRS,** H. Stone, May 1985. (PB85-213353/AS)
- 32 **Scatter Diagram and Histogram Program - SCATR,** H. Stone, May 1985. (PB85-213346/AS)
- 33 **TIMCHEK,** Gerald G. Rigdon, June 1985. (PB85-221257/AS)
- 34 **A MOS Temperature - PoP Forecast Plot,** William C. Randel, October 1985 (PB86-120029/AS).
- 35 **ROTODRAW,** Thomas Niziol, November 1985 (PB86-131828/AS)
- 36 **LAWEB: Data Processing for the Great Lakes,** William C. Randel and Matthew R. Peroutka, March 1986. (PB86-176658/AS)

NOAA SCIENTIFIC AND TECHNICAL PUBLICATIONS

The National Oceanic and Atmospheric Administration was established as part of the Department of Commerce on October 3, 1970. The mission responsibilities of NOAA are to assess the socioeconomic impact of natural and technological changes in the environment and to monitor and predict the state of the solid Earth, the oceans and their living resources, the atmosphere, and the space environment of the Earth.

The major components of NOAA regularly produce various types of scientific and technical information in the following kinds of publications:

PROFESSIONAL PAPERS — Important definitive research results, major techniques, and special investigations.

CONTRACT AND GRANT REPORTS — Reports prepared by contractors or grantees under NOAA sponsorship.

ATLAS — Presentation of analyzed data generally in the form of maps showing distribution of rainfall, chemical and physical conditions of oceans and atmosphere, distribution of fishes and marine mammals, ionospheric conditions, etc.

TECHNICAL SERVICE PUBLICATIONS — Reports containing data, observations, instructions, etc. A partial listing includes data serials; prediction and outlook periodicals; technical manuals, training papers, planning reports, and information serials; and miscellaneous technical publications.

TECHNICAL REPORTS — Journal quality with extensive details, mathematical developments, or data listings.

TECHNICAL MEMORANDUMS — Reports of preliminary, partial, or negative research or technology results, interim instructions, and the like.



Information on availability of NOAA publications can be obtained from:

**ENVIRONMENTAL SCIENCE INFORMATION CENTER (D822)
ENVIRONMENTAL DATA AND INFORMATION SERVICE
NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION
U.S. DEPARTMENT OF COMMERCE**

**6009 Executive Boulevard
Rockville, MD 20852**

